

(19) 대한민국특허청(KR) (12) 등록특허공보(B1)

(51) Int. Cl. ⁷ G05D 1/02		(45) 공고일자 (11) 등록번호 (24) 등록일자	2002년 11월 18일 10-0347192 2002년 07월 22일
(21) 출원번호 (22) 출원일자	10-1994-0012487 1994년 06월 03일	(65) 공개번호 (43) 공개일자	특 1995-0017684 1995년 07월 20일
(30) 우선권주장 (73) 특허권자	93-310932 1993년 12월 10일 일본(JP) 신코덴키 가부시기가이샤		
(72) 발명자	일본국 토오교우토 코우토우쿠 토오요 나나쥬우메 2반 14고 에가와다가미 일본국 미에쨩이세시다쨩가하나쥬 100반 지신꼬덴끼가부시끼가이샤이세세이샤 꾸쇼내		
(74) 대리인	이주기		

심사관 : 박재일

(54) 운행관리제어장치및그방법

요약

본발명은 공장등의 무인반송시스템에서의 무인반송차의 운행을 관리하고, 반송경로의 결정등을 행하는 운행관리제어장치 및 방법에 관한 것이다.

이는 복수 무인차의 운행을 관리하는 반송효율을 향상시키는 운행관리제어장치 및 방법을 제공하려는 목적을 가진다.

이를위해, 본발명은 경로탐색부(110)는 각각의 무인차의 코스트가 최소로 되는 주행경로(초기경로)를 구한다. 경로계획부(109)는 경로탐색부(110)에서 구해진 주행경로의 역주행구간을 조사하고, 그 결과에 기하여 역주행구간이 없는 기본경로를 구한다. 동작계획부(108)는 이 기본경로를 패쇄트리네트로 모델화하고, 그 시뮬레이션에 의한 통행불능이 검출된 경우에는 노드 통행순서의 조정 · 우회경로의 탐색 · 회피/대피경로의 탐색을 하여 주행불능을 해소한다. 이들의 결과로 부터 각 무인차의 이동의 추이(동작계획)가 얻어지도록 구성하였다.

대표도

도1

명세서

도면의 간단한 설명

- 제 1도는 본 발명의 제 1실시예에서의 운행관리제어장치의 블록도,
- 제 2도는 경로계획을 이용한 트리(Tree)도,
- 제 3도는 경로계획부의 경로계획처리를 도시한 흐름도,
- 제 4도는 동작계획처리의 동작예를 도시한 운행도,
- 제 5도는 제 4도의 운행도를 모델화한 패쇄 트리 네트도,
- 제 6도는 동작계획부의 회피경로 탐색처리를 도시한 흐름도,
- 제 7도는 회피경로탐색처리의 동작예를 도시한 운행도,
- 제 8도는 동작계획부의 데드락(Dead lock)파악 처리를 도시한 흐름도,
- 제 9도는 데드락 파악 처리의 동작예를 도시한 운행도,
- 제 10도는 동작계획부의 발화순서 조정 처리를 도시한 흐름도,
- 제 11도는 발화순서 조정 처리의 동작예를 도시한 운행도(a) 및 예약 시퀀스(b),
- 제 12도는 동작계획부의 우회경로 탐색처리를 도시한 흐름도,
- 제 13도는 우회경로 탐색처리의 동작예를 도시한 운행도,
- 제 14도는 동작계획부의 대피경로 탐색처리를 도시한 흐름도,
- 제 15도는 대피경로 탐색처리의 동작예를 도시한 운행도,
- 제 16도는 동작계획부의 동작계획처리(메인처리)를 도시한 흐름도 ①,

제 17도는 동작계획처리(메인처리)를 도시한 흐름도 ②,
 제 18도는 동작계획처리(메인처리)를 도시한 흐름도 ③,
 제 19도는 동작계획부의 동작예를 도시한 패쇄 트리 네트도 ①,
 제 20도는 동작계획부의 동작예를 도시한 패쇄 트리 네트도 ②,
 제 21도는 당해 동작예의 동작결과를 도시한 노드예약 시퀀스(a), 운행계획도(b),
 제 22도는 운행관리제어장치의 동작예1에서의 초기설정(a), 기본경로(b) 및 최종경로(c)이고,
 제 23도는 동작예 1에서의 초기설정(a), 기본설정(b) 및 최종설정(c)을 도시한 운행도,
 제 24도는 동작예 1에서의 운행계획도,
 제 25도는 운행관리제어장치의 동작예 2에서의 기본경로(a) 및 최종경로(b)이고,
 제 26도는 동작예 2에서의 초기경로(a), 기본경로(b) 및 최종경로(c)를 도시한 운행도,
 제 27도는 당해 동작예 2에서의 운행계획도,
 제 28도는 무인반송 시스템의 시스템 구성도,
 제 29도는 종례예(제28도)의 주행로에서의 코스트도,
 제 30도는 종례예에서의 이동경로를 도시한 운행도 ①,
 제 31도는 종례예에서의 이동경로를 도시한 운행도 ②,
 제 32도는 종례예에서의 이동경로를 도시한 운행도 ③이다.

* 도면의 주요부분에 대한 부호의 설명

101----반송로	102----운행관리제어장치
103----운행제어 데이터 메모리	104----반송지시 테이블메모리
105----무인차 데이터 메모리	106----주행로 데이터 메모리
107----제어부	108----동작계획부
109----경로계획부	110----경로탐색부

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 공장등의 무인반송시스템에 있어서, 무인반송차의 운행을 관리하고, 반송경로의 결정등을 행하는 운행관리제어장치 및 그 방법에 관한것이다.

제 28도는 복수의 무인차를 가지는 자동반송시스템의 시스템 구성도이다. 본도에 있어서, 운행관리제어장치(100)는 무인반송시스템의 관리를 행하고, 낱하형의 주행로(101)와 무인차 #1, #2.....#5가 있다. 또한, 1, 2,, 28은 주행로(101)상에 점재(點在)하는 노드이고, 무인차 #1~#5는 이들 노드에서의 정지·방향전환 및 반송물의 적치 하강 작업을 행한다. 또한, 무인차 #1~#5의 각각은 목표지점(이하, 목표노드)까지의 주행 경로를 결정하는 기능을 가지고, 운행관리제어장치(100)에 의해 부여되는 목표노드까지 자발 결정한 경로에서 이동을 향한다. 더구나, 이 경로의 결정에 대해서는 최적경로결정장치(일본 특원평 5-77244호)에 상술되어 있다.

이하에서 이 무인반송시스템의 동작예를 설명한다.

먼저, 운행관리제어장치(100)에서 무인차 #1~#5로, 제 28도에 도시한 이동지시가 송부된다고 하자. 그러면, 무인차 #1~#5는 지시된 이동목표까지의 최적의 주행경로를 작성한다. 이 퇴적경로의 결정에는 인접노드를 묶는 각 주행구간의 코스트가 사용되고, 이 코스트의 적산치가 가장 작은 경로가 선택된다. 단, 이 주행경로의 작성에 있어서는 타 무인차의 주행경로는 고려되어 있지 않고, 타 무인차가 존재하지 않는 경우에만 최적주행경로로 된다. 또한, 여기서 말하는 코스트란 각 주행구간의 통과에 요하는 시간 등의 지표이다.

제 29도는 주행로(101)에서의 코스트를 도시한 도면이고, 본 도에서의 각 주행구간의 코스트는 ()내에 기재되어 있다.

제 30도는 이때 작성된 각 무인차 #1~#5의 주행경로를 도시한 도면이고, 본도의(a)는 각 무인차 #1~#5의 주행경로를 실선·점선·파선·일점쇄선·2점파선으로 각각 도시한 운행도이고, 동도(b)는 이들의 경로를 노드열로 도시한 도면이다.

다음에 무인차 #1~#5는 자신의 주행경로상의 노드번호를 이동순으로 운행관리제어장치(100)로 보내고, 노드의 예약을 행한다. 운행관리제어장치(100)는 요구된 노드열{제 30도 (b)}을 최초부터 순서대로 조사하고, 타 무인차가 예약하고 있지않은 경우에는 그 예약을 허가한다. 그리고, 무인차 #1~#5는 허가된 노드까지 이동을 한다. 이들 제어에 의해 무인차간의 충돌이 방지된다.

지금, 예를들어, 무인차 #1~#5가 노드 4, 6, 20, 22, 3까지 각각 진입하였다고 하자. 제 31도는 이때의 각 무인차 #1~#5의 현재위치 및 그후의 주행경로를 도시한 운행도이다. 이 경로에서 그냥 다음의 이동이 행해지면, 무인차 #1 및 #2는 동일주행로를 서로 역방향으로 이동한다고 하는 주행로의 경합이 발생한다. 이러한 경우 어느쪽인가가 경로를 바꾸지 않는한 목적지에 도달할 수 없다. 또한, 이때 무인차 #1 및 #2는 다음 이동처의 노드의 예약이 허가되지 않는다.

여기서, 무인차 #1이 우회로(노드 4→ 18→ 19→ 20→ 21→ ...)를 찾은 노드 18의 예약을 행한다. 이에 의해 무인차 #2는 당초의 경로로 이동을 행할수 있지만, 이번은 무인차 #1 및 #3 간의 주행로의 경합이 발생한다. 제 32도는 이때의 각 무인차 #1~#5의 경로를 도시한 운행도이다. 이번은 무인차 #3이 우회로(노드 20→ 6→ 5→ 4→ 3→ 16→ 15)를 찾고, 노드 6을 예약한후 그 노드로 이동을 한다.

이렇게 주행로의 경합과 우회로 탐색을 반복하고, 각 무인차 #1~#5는 목적지점까지 이동을 한다.

그러나, 이상의 방법은 주행로의 경합이 발생하고부터 이동 경로를 변경하기 때문에, 우회를 반복하는 등의 쓸모없는 이동이나 대기가 발생하는 경우가 있고, 무인차의 수량이 증가함에 따라 반응효율이 대폭 저해되어 버린다는 문제가 있다.

본발명은 이러한 배경하에 이루어진 것이므로 복수의 무인차가 효율좋게 목표지점까지 이동을 할 수 있는 운행관리제어장치 및 그 방법을 제공하는것을 목적으로 하고 있다.

청구항 1기재의 발명은 정지위치에 있는 복수의 노드와, 이들노드간을 접속하는 접속로로 되는 주행로를 주행하는 복수 무인차의 운행을 제어하는 운행관리제어장치에 있어서, 각 무인차가 동일접속로를 서로 역방향 주행하지 않도록 각 무인차의 주행경로를 탐색하는 제 1수단과, 제 1수단에 의해 탐색된 각 무인차의 주행경로에 기하여 각 무인차의 시간적인 이동을 시뮬레이션(Simulation)하고, 어느 것인가 무인차의 진행불능을 검출한 경우에 주행순서변경·우회경로탐색·대피경로탐색의 어떠한 방법으로 무인차의 진행불능을 해제하는 제 2수단과를 구비하여 되는 것을 특징으로 하고 있다.

청구항 2기재의 발명은 정지위치에 있는 복수의 노드와, 이들 노드간을 접속하는 접속로로 되는 주행로를 주행하는 복수 무인차의 운행을 제어하는 운행관리제어방법에 있어서, 각 무인차가 동일 접속로를 서로 역방향주행하지 않도록 각 무인차의 주행경로를 탐색하는 제1 스텝과, 제1 스텝에 의해 얻어진 주행 경로에 기하여 각 무인차의 시간적인 이동을 조사하고, 어느 것인가 무인차의 진행불능이 검출된 경우에는 이미 주행을 종료하고 있는 무인차의 경로에 다른 노드로 후퇴하는 퇴피(退避)경로를 추가하는 제 2 스텝과, 제 2스텝에서의 진행불능이 해소될 수 없는 경우에 무인차의 주행순서를 변경하는 제 3스텝과, 제 3스텝에서의 진행불능이 해소될수 없는 경우에 무인차의 경로에 우회경로를 추가하는 제 4스텝과, 제4 스텝에서의 진행불능이 해소될수 없는 경우에 무인차의 경로에 일시적으로 다른 노드에서 후퇴해 다시 원래 노드로 돌아가는 대피경로를 추가하는 제 5스텝과를 가지는 것을 특징으로 하고 있다.

청구항 1기재의 발명에 의하면, 제 2수단은 제 1수단에 의해 얻어진 주행경로에 기하여 각각의 무인차의 시간적인 이동을 시뮬레이션하고, 그 경로에서 무인차의 진행불능을 검출한 경우에는 주행순서 변경·우회경로탐색·대피경로탐색등을 하여 통행 불능을 해제하기 때문에 이동이 지체되지 않는 주행경로및 주행 순서를 얻을수 있고, 따라서 이동효율을 향상시킬수 있다는 효과가 얻어진다.

청구항 2기재의 발명에 의하면, 무인차의 진행불능이 검출된 경우에는 퇴피경로의 추가·주행순서의 변경·우회경로의 추가·대피경로의 추가의 순으로 진행불능을 해제하기 때문에 모든 무인차의 이동이 지체되지않는 경로 및 주행순서를 얻을수 있고, 따라서 이동효율을 향상시킬수 있다는 효과가 얻어진다.

실시예

이하 도면을 참조하여 본발명의 1실시예에 대하여 설명한다.

제 1도는 본실시예에 의한 운행관리제어장치(102)의 구성을 도시한 블록도이다. 본도에 있어서 운행제어 데이터메모리(103)는 각 무인차의 이동경로·후술하는 노드 예약시퀀스(Sequence) 등을 기억한다. 반송지시데이블메모리(104)는 반송물의 위치와 반송처등을 기억한다. 무인차데이터메모리(105)는 각 무인차의 현재위치·이동방향등의 상태를 기억한다. 주행로데이터메모리(106)는 주행로상의 각 노드의 좌표와 그 접속관계 및 코스트(Cost)등을 기억한다(제 29도). 또한, 제어부(107)는 무인차의 최적한 주행경로 및 동작순서를 결정하는 제어부이다. 이 제어부(107)는 CPU등에 의해 구성되고, 기능적으로는 동작계획부(108)·경로계획부(109)·및 경로탐색부(110)로 나눌수 있다.

동작계획부(108)·경로계획부(109) 및 경로탐색부(110)에 대해서는 아래에 상술한다.

A : 경로탐색부(110)

먼저 경로탐색부(110)는 경로계획부(109)로 부터 공급되는 경로탐색지시에 따라 각 무인차의 출발노드 및 목표노드를 묶은 경로를 모두 구한다. 다음으로, 주행로데이터메모리(106)에 기억된 코스트(제 29도)로 부터 각 경로의 코스트를 각각 적산하고, 그 코스트가 최소로 되는 경로를 최적경로로 선택한다. 단, 경로탐색지시에 후술하는 방향결정의 방향정보가 포함된 경우에는 방향결정된 마크(Mark)를 역방향으로 주행하는 경로는 선택되지 않는다. 동일하게, 경로탐색지시에 후술하는 통행금지의 방향정보가 포함되는 경우에는 통행금지의 마크를 통하는 경로는 선택되지 않는다.

이상의 방법으로 구한 경로 및 그 코스트는 경로계획부(109)로 출력된다. 단, 여기서 작성된 경로는 다른 무인차의 주행경로는 고려되지 않고, 주행로의 경합이 없는 경우에만 최적한 경로로 된다.

B : 경로계획부(109)

경로계획부(109)는 트리(Tree)의 탐색수법을 사용하여 역주행구간이 없는 주행경로를 구해 그 결과를 동작계획부(108)로 출력한다.

여기서 말하는 트리는 제 2도에 도시한 바와 같이 아래쪽으로 나뉘어서 분기를 하는 구성을 취한다. 여

기서, N1·N2····는 분기조건이 가해진 분기점이고, 이중 분기점 N1은 분기를 개시하는 루트(Root)분기점이다. 또한, 예를들어, 분기점 N2를 현재의 분기점으로 하면, 분기점 N1은 분기점 N2의 모분기점(母分岐点)으로 되고, 분기점 N3 및 N4는 분기점 N2의 자분기점(子分岐点)으로 된다. 탐색은 기본적으로 상위의 분기점에서 하위의 분기점에 걸쳐서 행하여지지만, 탐색불능의 경우에는 일단 모분기점으로 돌아가(이하, 백트랙(Back Track)이라 한다) 다른 분기점으로 분기한다.

제 3도는 경로계획부(109)가 행하는 경로계획처리를 도시한 플로우차트이고, 본도를 기본으로 아래에 설명한다.

처리가 개시되면(스텝 SP1) 스텝 SP2에 있어서 경로탐색부(110)로 탐색지시를 내고, 각 무인차의 주행경로를 구한다. 여기서, 탐색지시를 받은 경로탐색부(110)는 상기한 방법에 의해 경로를 탐색하고, 그 결과인 초기경로를 경로계획부(109)로 출력한다. 더욱이, 이 탐색지시에는 반송지시테이블메모리(104)에 기억된 데이터에 의해 결정되는 무인차의 목표노드가 포함된다.

스텝 SP3에서는 트리의 루트분기점을 비워둔다.

스텝 SP4에서는 경로탐색부(110)에서 공급되는 각 무인차의 주행경로에 기하여 임의의 두개의 무인차가 서로 역방향으로 이동을 행하는 구간(역방향 구간)을 구하고, 이를 무인차 전체조합에 대하여 행한다.

스텝 SP5에서는 스텝 SP4의 결과에서의 역방향 구간이 없으면 처리를 종료하고(스텝 SP17), 역방향 구간이 있는 경우에는 스텝 SP6으로 진행한다. 또한, 역방향 구간이 없는 경우에는 그때의 주행경로가 최종적인 주행경로로 된다.

스텝 SP6에서는 각 무인차의 경로의 역방향 구간의 코스트를 적산한다. 여기서, 역방향 구간의 코스트는 주행로데이터메모리(106)로부터 읽히어진다. 또한, 어느 역방향 구간에서 다른 복수 무인차의 경로와 역방향 경합을 이루고 있는 경우에는 그 경합의 회수분 코스트를 적산한다. 단, 주행로상의 서로 인접하는 2점간을 묶는 경로가 그 외에는 존재하지 않는 경우에는 그 구간은 역방향 구간에 포함되지 않는다.

스텝 SP7에서는 각 무인차에 붙여진 부호를 역주행 구간 코스트의 큰 순으로 정렬한 경합 무인차집합을 작성한다.

스텝 SP8에서는 이 경합 무인차집합을 가진 분기점을 모분기점의 아래에 붙였다. 단, 이 스텝 SP8이 처음 처리된 경우에는 루트분기점에 상기 경합 무인차집합을 설정한다.

스텝 SP9에서는 경합 무인차집합에서 주목무인차를 결정한다. 이 주목무인차는 코스트가 큰 순으로 정렬된 경합 무인차집합의 최초 무인차로부터 순차 선택되 온다. 또한, 다음 무인차가 없는 경우에는 주목무인차 없음으로 한다.

스텝 SP10에서는 전 스텝 SP9의 처리에서의 주목무인차가 없었던 경우에는 다음 스텝 SP11로 진행하고, 주목무인차가 있는 경우에는 스텝 SP13으로 분기한다.

스텝 SP11에서는 현재의 분기점이 루트분기점 전에 있는가 아닌가를 조사하고, 루트분기점이 아닌 경우에는 다음의 스텝 SP12로 진행하고, 루트분기점인 경우에는, 즉 루트분기점의 경합 무인차집합 모두에 있어서 경로 정리가 실패한 경우, 경로정리 실패로 모든 처리를 종료한다(스텝 SP17).

스텝 SP12에서는 현재분기점의 처리를 모분기점으로 옮김(백 트랙)과 함께, 스텝 SP9의 처리로 돌아간다. 또한, 현재의 분기점으로 분기할때에 행했던 방향결정은 이때에 해제한다.

스텝 SP13에서는 주행로내에 주목무인차 경로의 역방향구간을 동 무인차의 이동방향의 역방향에 방향결정하고(일방통행으로 한다), 방향정보에 가산한다. 스텝 SP14에서는 경로탐색부(110)로 탐색지시를 내고, 이 방향 결정된 주행로에서의 모든 무인차의 경로를 구해 고친다.

스텝 SP15에서는 전 스텝 SP14의 경로탐색에서 구해지지 않은 경로가 존재하는가를 조사하고, 존재하는 경우에는 다음 스텝 SP16으로 진행하고, 존재하지 않는 경우에는 스텝 SP4로 돌아간다.

스텝 SP16에서는 스텝 SP13에서 행해졌던 주행로의 방향결정을 해제한 후 스텝 SP9로 돌아간다.

이상 설명한 처리에 의해 코스트가 작고 역주행 구간이 없는 복수의 무인차 경로(이하, 기본경로)가 얻어진다.

C : 동작계획부(108)

동작계획부(108)는 경로계획부(109)에서 공급되는 기본경로에 기하여 각 무인차의 이동을 시간적으로 조사하고, 이동 순서를 조정하거나, 필요에 의하여 경로를 변경·추가하는 등을 하여 모든 무인차의 목표노드까지의 이동동작을 계획한다. 또한, 그 계획은 패쇄트리네트를 이용한 시뮬레이션에 의해 행하여진다. 동작계획부(108)가 하는 동작계획 처리를 설명하기 전에 패쇄트리네트 및 동작계획에 포함되는 각종 처리의 설명을 구체예를 들어 행한다.

1. 패쇄트리네트

제4도(a)는 패쇄트리네트의 설명에 이용되는 운행도이고, 본 도에 있어서 주행로(111)의 노드(2,6)위에는 무인차 #1·#2가 각각 대기하고 있다. 또한, 제4도(b)는 무인차 #1·#2의 각각 출발노드·목표노드 및 목표노드까지의 경로를 도시한 도면이고, []안에는 노드간의 이동시간(초)을 표시한다. 즉, 무인차 #1은 노드2에서 노드3·4의 순으로 이동을 하고, 각각의 이동시간은 노드2에서 노드3까지가 1초, 노드3·4간이 3초이다. 또한, 무인차 #2에 대해서도 동일하다.

제 5도는 상기한 운행도 {제 4도(a)}를 패쇄트리네트로 모델화한 도면이다. 본도에서 P1·P2····P8는 각각 플레이스(Place)이고, 주행로(111)의 각 노드1내지 8에 대응하고, 각 노드의 정유상태를 나타낸다. 또한, 이들 플레이스 P1 내지 P8에는 대응하는 노드에 무인차가 있는 경우에는 원내

에 흑점이 표시되고, 노드가 예약되어 있는 경우에는 흰 원상태로 표시된다. 초기 상태에서는 무인차 #1은 노드2에 · 무인차 #2는 노드 6에 있기 때문에 플레이스 P2 · P6에 각각 검은 점이 표시된다.

또한, T12 · T23 · · · 는 트랜지션(Transition)이고, 무인차의 이동 상태를 나타낸다. 또한, 동 트랜지션에는 그것에 입력된 입력마크와 그로부터 출력되는 출력마크가 1개씩 붙어있고, 이들 마크에 의해 인접하는 2개의 플레이스가 묶여진다. 예를 들면, 입력노드5 · 출력노드6의 트랜지션은 T56이고, 역으로, 노드6에서 노드5로의 이동에 대응하는 트랜지션은 T65이다. 또한, 노드5에서 노드6으로의 이동시에는 이 트랜지션 T56을 발화시켜 무인차가 이동중에 있는 것을 나타낸다. 또한, 트랜지션은 한번 발화하면 대응하는 마크의 이동시간등에 기하여 유한시간 그 발화를 지속한다.

또한, 이로부터 이동을 하는 경로상의 트랜지션을 그 발화순으로 정렬한 것을 발화예정 트랜지션계열로 칭하고, 제4도(b)의 기본경로의 경우 무인차 #1 및 #2의 발화예정 트랜지션계열은 다음과 같이 된다.

발화예정 트랜지션계열(무인차 #1) = {T23 · T34}

발화예정 트랜지션계열(무인차 #2) = {T67 · T73 · T34}

다음에 트랜지션 Tst의 발화에 관한 처리를 설명한다.

① 발화가능조건

트랜지션 Tst는 입력측 플레이스 Ps에 흑점이 있고, 출력측 플레이스 Pt에 흑점이 없고, 더구나 이에 앞서 선행트랜지션(후술함)이 모두 발화하고 있을때 발화가능으로 된다.

② 발화처리

발화가능한 트랜지션 Tst가 발화하면 노드 Ns에서 노드 Nt까지의 이동시간을 현재시간에 가산하고, 트랜지션 번호도 함께 그 무인차의 완료시각에 세트한다. 또한, 노드 Nt에서 작업하는 경우에는 작업시간을 또한 완료시각에 가산한다. 그래서, 출력플레이스 Pt에 흰원을 둔다.

③ 발화완료처리

발화중의 트랜지션 Tst의 입력측 플레이스 Ps에서 검은점 · 출력측 플레이스 Pt에서 흰원을 각각 삭제하고, 플레이스 Pt에 검은점을 둔다.

더우기, 상술했던 발화가능조건에서의 무인차가 단선구간(우회로가 없는 구간)을 통과할때는 트랜지션을 일괄하여 발화가능이 조사된다. 예를들면, 발화예정 트랜지션계열 {T1 · T2 · · · Tn} 이고, 이중 트랜지션 $T_i \cdot T_{i+1} \cdot \cdot \cdot T_j$ 가 단선구간에 있는 경우 트랜지션 T_{i-1} 은 트랜지션 $T_i \cdot \cdot \cdot T_j$ 가 모두 발화가능한 경우에만 발화가능하다. 이것은 트랜지션 T_{i-1} 의 발화에 의해 단선구간에 존재하는 다른 무인차의 출구를 폐쇄하지 않도록 하기 때문이다.

II. 퇴피(退避)동작의 계획

퇴피동작은 이동중에 있는 무인차의 이동처에 작업을 끝내고 대기상태에 있는 다른 무인차가 존재하는 경우에는 그 대기상태의 무인차를 다른 노드에 이동(퇴피)시키는 동작이다. 제 6도는 이 퇴피경로를 찾아내는 퇴피경로 탐색처리의 플로어차트이고, 아래에 그 설명을 한다.

퇴피동작이 개시되면 스텝Sa2에서의 자신의 주변노드를 조사하고, 퇴피가능노드를 모두 구한다. 여기서 퇴피가능노드란 아래의 조건을 만족하는 노드이다.

① 그 노드로의 이동의 금지되어 있지 않음.

② 대기상태가 아닌 무인차에 점유되어 있지 않음.

다음에 스텝Sa3에서의 퇴피가능노드에 대하여 이동시간등의 기본코스트를 적산한다. 단, 대기상태의 무인차가 있는 노드에 대해서는 예를들어 코스트를 100배 하는등 하여 되어지는 것을 선택하지 않도록 한다.

스텝Sa4에서는 스텝Sa3의 결과중 코스트가 가장 작은 노드를 퇴피노드로 하고, 현재노드에서 퇴피노드까지의 구간을 무인차의 주행경로에 추가한다.

제7도는 이상의 퇴피경로탐색의 1예를 도시한 운행정도이다. 본 도에서의 무인차 #1 및 #3은 대기중(이동 예정없음)이고, 무인차 #2는 노드3으로 이동하려고 하고 있다. 이 경우 무인차 #2의 방해가 되는 무인차 1이 퇴피의 대상으로 된다. 이 무인차 #1의 대피가능노드는 노드2 및 노드7이고, 그 양 노드로의 퇴피경로의 코스트를 조사한다. 노드2로의 이동시간은 1초이지만 무인차 #3이 노드2에서 대기중이기 때문에 기본코스트는 그 이동시간의 100배인 100으로 된다. 또한, 노드7로의 이동시간은 4이고, 기본코스트도 4로 된다. 따라서, 코스트가 작은 노드7이 퇴피노드로 선택되고, 노드3→ 7이 무인차 #1의 경로에 추가된다.

III. 데드락(Dead Lock)

복수의 무인차가 좁은 영역에 밀집해버려 상기한 퇴피동작을 행해도 예정했던 경로에서 전진할 수 없는 데드락의 상황에 빠진 경우에는 후술하는 발화순서조정 · 우회경로탐색 · 대피경로탐색이 행해진다. 여기에서는 데드락의 상황을 파악하는 방법에 대하여 기술한다. 제 8도는 이 데드락 파악처리를 도시한 플로어차트이고, 아래에 이 설명을 한다.

먼저, 스텝Sb2에서의 다시 아래의 처리를 행하지 않는 무인차를 택하고 그 무인차가 대기중에 있는가를 조사한다. 만약, 대기중의 무인차가 아닌 경우에는 스텝Sb3으로 진행하고, 대기중에 있는 경우에는 스텝Sb4로 분기한다.

스텝Sb3에서는 대상으로 하고 무인차의 주행경로상에 있는 가장 가까운 무인차를 무용한 무인차로 하

고, 스텝Sb5로 진행한다.

스텝Sb4에서는 대상으로하고 있는 무인차의 주변에 있는 모든 무인차를 무용한 무인차로 하고, 다음 스텝Sb5로 진행한다.

스텝Sb5에서는 모든 무인차에 대하여 상기의 처리가 종료했는가를 조사하고, 다시 처리않된 무인차가 있는 경우에는 스텝Sb2로 돌아간다.

스텝Sb6에서는 적당한 대기중에 있지 않은 무인차를 고르고, 그 무인차가 방해하고 있는 무인차, 그 방해가 되어있는 무인차가 방해하고 있는 무인차,로 그와같이 더듬어 가고, 그 안에서 2대 이상의 루우프를 찾는다. 여기서, 이를 모든 조합에 대해서 행한다.

스텝Sb7에서는 얻어진 루우프중 가장 많은 무인차를 포함하는 루우프를 경합루우프로 선택한다.

스텝Sb8에서는 스텝Sb7의 처리에서 경합루우프가 얻어지지 않을 때에는 데드락 파악실패로 본처리를 종료한다(스텝Sb10).

스텝Sb9에서는 경합루우프의 각 무인차에 대하여 그 무인차가 움직이는 인접노드 즉, 다른 무인차가 없는 노드를 구하고, 본처리를 종료한다(스텝Sb10).

제 9도는 주행로(101)에서의 데드락의 1예를 도시한 도면이고, 본 도의(a)는 그 초기상태의 운행도, (b)는 데드락상태의 운행도를 도시하고 있다. 또한, 본도에서의 대기중이 아닌 무인차는 편상(編上)하여 도시하고, 대기중의 무인차는 하얗게 도시한다. 여기서, 무인차 #1 내지 #6의 이동경로는 다음과 같다.

무인차 #1 : 6→ 5→ 4

무인차 #2 : 5→ 4

무인차 #3 : 17→ 18→ 4

무인차 #4 : 2→ 3→ 16

무인차 #5 : 16→ 17

무인차 #6 : 15→ 16→ 17

상기의 경로에 따라서 무인차 #1 내지 #6이 제 9도(a)의 초기경로에서 1구간만 이동하면 동도(b)에 도시한 바와같이 데드락의 상태로 된다. 여기서, 무인차 #2 및 #5는 대기상태에 있지만, 상술했던 퇴피경로는 보이지 않는다. 그래서, 무인차 #1 내지 #6의 각각이 방해하는 무인차를 조사하고, 제9도(c)에 도시한 바와같은 결과를 얻는다. 본도에서 예를들어 무인차 #1이 무효로 되는 것은 무인차 #2에서이고, 데드락시에 무인차 #1의 이동가능한 빈 노드는 노드6이다. 이 결과를 기준하여 방해하는 무인차 혹은 방해로 되는 무인차의 루우프를 탐색하면,

루우프 1 : 무인차#1(5) → #2(4)

루우프 2 : 무인차#3(18) → #2(4)

루우프 3 : 무인차#6(16) → #5(17)

루우프 4 : 무인차#3(18) → #2(4) → #4(3) → #6(16) → #5(17) → #3의 4개의 루우프가 얻어진다. 그래서, 이중 가장 많은 무인차를 포함하는 루우프4가 경합루우프로 선택된다. 또한, ()내에는 데드락시의 각 무인차의 현재위치(노드)이다.

IV. 발화순서 조정

상기한 데드락에 빠진경우, 먼저, 트랜지션의 발화순서를 조정하고, 그 해소를 시도한다. 즉, 노드의 예약순서를 바꿈에 의해 데드락이 회피가능하지 않은가를 조사한다. 제 10도는 이 발화순서조정처리를 도시한 플로우차트이고, 아래에 그 설명을 한다.

먼저, 스텝 Sc 2에서는 경합 루우프에 속하는 대기하지 않는 각 무인차의 현재노드를 다른 무인차가 이후 통과하는 횟수를 헤아린다.

스텝 Sc 3에서는 상기 통행횟수가 「0」 인가를 조사하고, 「0」 인 경우에는 본처리를 종료하고(스텝 Sc 9), 「0」 이 아닌 경우에는 다음 스텝 Sc 4로 진행한다.

스텝 Sc 4에서는 각 노드의 통행횟수가 이전회의 발화순서조정처리시와 다른 새로운 상태로 있는가를 조사하고, 새로운 상태라면 다음 스텝Sc5로 진행하고, 이 전회와 같은 상태라면 본 처리를 종료한다(스텝Sc9). 이것은 상태가 복잡해지면 발화순서를 몇번이고 바꾸어서 결국 전상태로 돌아가기 때문이다.

스텝 Sc 5에서는 거기에 있는 무인차는 자신을 방해로 하고 있는 무인차는 현재 노드를 통과하고 있는가를 조사하고, 통과하지 않은 경우에는 다음 스텝 Sc 6으로 진행하고, 통과하는 경우에는 처리를 종료한다(스텝 Sc 9). 이것은 동일구간을 통과해온 무인차에서는 이후에 온 무인차를 앞에 두지 않도록 하기 때문이고, 합류점(노드)을 체크하기 때문이다.

스텝 Sc 6에서는 스텝 Sc 2의 결과인 통행횟수를 평가치로 한다. 단, 자신을 방해로 하고 있는 무인차가 대기상태인 때에는 평가치를 올린다. 또한, 이상의 스텝 Sc 2~Sc 6의 처리는 경합 루우프에 속하는 대기하지 않는 무인차가 있는 모든 노드에 대해서 행해진다. 스텝 Sc7에서는 스텝 Sc6에서 구한 각 노드의 평가치가 최소로 되는 노드를 경합노드로 선택한다.

스텝 Sc 8에서는 그 경합노드에 있는 무인차를 방해로 하는 무인차가 먼저 통과하도록 트랜지션 발화제어데이터에 추가한다. 이 트랜지션 발화제어데이터는 특정 노드에서 무인차의 이동순서를 규제하는 것이

고, 이에 의해 트랜지션의 발화가 규제된다.

이상에서 설명한 발화순서조정 처리를 상기한 제 9도의 실행도를 이용하여 설명한다. 먼저, 경합루우프중의 대기하지 않는 무인차 #3 · #4 · #6이 있는 노드 18 · 3 · 16 {제 9도(b)}의 각각에 대해서 통행횟수를 구한다. 이 결과, 노드 16이 1회 · 다른 노드는 0회이기 때문에 노드 16이 경합노드에 뽑힌다. 이 노드 16을 출력선에 가지는 트랜지션은 트랜지션 T15 · 16 및 T3 · 16이고, 이들의 발화순서를 역으로 한다. 즉, 트랜지션 T3 · 16을 선행트랜지션으로 하고, 트랜지션 T15 · 16보다도 먼저 발화시킨다. 이에 의해 무인차 #6을 노드 16으로 이동시키기 전에 무인차 #4를 노드 16까지 이동시키고, 노드 3이 비기때문에 대기중인 무인차 #2를 노드 3으로 퇴피시킬 수 있다. 이렇게, 트랜지션의 발화순서를 조정하는 것으로 데드락을 해소가능한 경우가 있다.

제 11도는 이 구체예의 최종결과를 도시한 도면이고, 본도(a)는 무인차 #1 내지 #6의 출발노드에서 목표노드까지의 경로를 도시하고, 본도(b)는 노드예약 시퀀스를 도시한다. 본도 (a)의 ()안은 목표노드이고, 이 노드 이하의 경로는 상술한 퇴피경로이다. 또한, 노드예약 시퀀스 {본도 (b)}는 각 노드를 무인차가 예약하는 순서를 도시한 것인데, 본도에서의 예를들면 노드 5는 무인차 #2 · #1의 순으로 예약이 행해진다.

V. 우회동작의 계획

기본 경로에 따라서 데드락의 상황을 해소할 수 없는 경우에는 적당한 무인차가 우회경로를 취하도록 계획한다. 제 12도는 이 우회경로 탐색처리를 도시한 플로차트이고 아래에 이 설명을 한다.

먼저, 스텝 Sd2에서의 경합루우프에 속하고, 대기중이 아닌 무인차에 대하여 이동가능한 인접노드가 있는가를 조사한다. 있는 경우에는 다음의 스텝 Sd 3으로 진행하고, 없는 경우에는 우회경로탐색 실패로 본처리를 종료한다(스텝 Sd 9).

스텝 Sd 3에서는 앞 스텝에서 뽑았던 무인차에 관하여 아래의 마크를 일시적으로 통행금지로 한다.

① 그 무인차의 현재노드에서 다음노드로의 마크

② 현재 노드에서 움직이지 않는 방향으로의 마크

스텝 Sd 4에서는 현재 노드를 출발노드 · 다음노드를 목표노드로 설정한다.

스텝 Sd 5에서는 스텝 Sd 3 및 4의 설정에 기하여 경로탐색처리(스텝 Sa 1)를 행한다.

스텝 Sd 6에서는 스텝 Sd 3에서 하였던 마크의 통행금지를 해제한다.

스텝 Sd 7에서는 스텝 Sd 5의 경로탐색처리에서의 우회경로가 구해지는가를 조사하고, 우회경로가 있는 경우에는 다음의 스텝 Sd 8로 진행하고, 우회경로가 없는 경우에는 우회경로탐색 실패로 본처리를 종료한다(스텝 Sd 9).

스텝 Sd 8에서는 대상으로 하고 있는 구간의 경로를 우회경로로 교체하여 본처리를 종료한다(스텝 Sd 9).

제 13도는 데드락의 상황을 예시한 도면이고, 본도(a)는 데드락의 상황을 도시한 실행도이다. 이하, 본도에 기하여 상술한 우회경로탐색처리를 설명한다.

먼저, 무인차 #1이 최초로 뽑혔다고 하면, 노드 4→ 3 · 4→ 18의 마크를 통행금지로 하여 우회경로를 구하도록 한다. 이 경우의 무인차 #1의 우회경로로서는 노드 5→ 6→ 20→ 19→ 18→ 17···의 경로가 고려되지만, 노드 18→ 17이 무인차 #4의 경로와 역행하기 때문에 기본적으로는 이 경로는 선택되지 않는다. 단, 후술하는 동작계획(제 16도)의 2회째의 시행에서는 역주행 구간이 통행금지로 되지 않기 때문에 뽑혀질지도 모른다.

다음에, 무인차 #2에서 우회경로탐색을 하면 노드 3→ 2→ 1→ 15→ 16이 얻어진다. 그래서, 이 경로가 우회경로로 땀과 함께 무인차 #2의 경로(3→ 16)의 사이에 삽입된다(제 13도(b)).

VI. 대피동작의 계획

데드락의 상황에서의 우회동작이 취해지지 않는 경우는 적당한 무인차가 일단 별도의 노드로 후퇴하고(대피), 다른 무인차에 도로를 양보한 후 다시 원래의 경로에서 이동을 한다. 제 14도는 이 대피동작을 도시하는 대피경로탐색이고, 아래에 이 설명을 한다.

먼저, 스텝 Se 1에서는 경합루우프에 속하고, 또 목표노드까지 도착하지 않은 무인차(대기하지 않는 무인차)에 대하여 이동 가능 인접노드가 있는가를 조사하고, 있는 경우에는 다음의 스텝 Se 3으로 진행하고, 없는 경우에는 대피경로탐색 실패로 본 처리를 종료한다(스텝 Se 10).

스텝 Se 3에서는 그 무인차의 현재 노드에서 다음 노드로의 마크를 일시적으로 통행금지로 한다.

스텝 Se 4에서는 현재 노드를 출발 노드로 · 다른 모든 노드를 목표노드로 설정한다.

스텝 Se 5에서는 스텝 Se 3 및 4에서의 설정된 조건으로 경로탐색처리(스텝 Sa 1)를 한다.

스텝 Se 6에서는 스텝 Se 5에서 얻어진 모든 경로중 가장 코스트가 작은 경로를 선택하고, 그 목표노드를 대피노드로 한다. 단, 대피노드의 선택으로는 단선구간에 존재하는 노드를 제외한다.

스텝 Se 7에서는 스텝 Se 3에서 행했던 통행금지를 해제한다.

스텝 Se 8에서는 스텝 Se 6의 결과에서 대피노드가 없는 경우에는 대피경로탐색 실패로 본처리를 종료하고(스텝 Se 10), 대피노드가 있는 경우에는 다음의 스텝 Se 9로 진행한다.

스텝 Se 9에서는 현재 노드에서 대피노드, 더우기 거기로부터 현재 노드까지의 경로(대피경로)를 현재

대기하고 있는 경로에 삽입하고, 본처리를 종료한다.

제 15도(a)는 데드락의 상황을 예시한 운행정도이고, 본도에 기하여 대피경로탐색처리를 설명한다.

먼저, 무인차 #1이 최초로 뱉혀졌다하면 노드 4→3을 일시 통행금지로 하여 경로탐색을 하고, 노드 5가 가장 코스트가 작은 대피노드로 선택된다. 여기서 이 무인차 #1의 이동가능한 인접노드는 노드 5이외에 노드 18이 있지만, 코스트(제 29도 참조)가 작은 노드 5가 대피노드에 뱉혀진다. 다음에, 노드 5에서 노드 4로의 경로를 탐색하고, 노드 4→5→4가 대피경로로 구해진다. 최후에 이 대피경로를 원래 경로에 삽입한, 무인차 #1의 경로(노드 4→5→4→3→2)가 얻어진다 {제 15도(b)}.

VII. 동작계획

동작계획부(108)는 I~VI의 각종 처리를 이용하여 모든 무인차의 경로결정 및 동작순서의 계획을 한다. 제16도·17도·18도는 이 동작계획처리를 도시한 플로우차트이고, 아래에 이 설명을 한다.

먼저, 스텝 Sf 1(제 16도)에서의 패쇄 트리네트를 이용하여 주행로의 모델화를 한다.

스텝 Sf 2에서는 주행횟수에 1을 세트한다.

스텝 Sf 3에서는 각 무인차의 경로를 경로계획에서 얻은 기본 경로에 설정한다.

스텝 Sf 4에서는 시행횟수를 조사하고, 시행횟수가 「1」이 되면 다음의 스텝 Sf 5로 진행하고, 「1」이 외의 값이 되면 스텝 Sf 6으로 진행한다.

스텝 Sf 5에서는 상기한 기본경로의 각 경로와 역행하는 마크를 모두 통행금지로 한다. 이에 의해 아래의 처리에서 역방향구간이 발생하지 않게 된다. 또한, 루우프①에 의한 처리가 반복시행횟수가 2(제 18도, 스텝 Sf 31)로 되는 경우에는 이 스텝 Sf 5는 실행되지 않고, 경로탐색처리에서의 이 통행금지의 제한은 증가되지 않는다.

스텝 Sf 6에서는 현재시각을 0으로 초기설정한다.

스텝 Sf 7에서는 각 무인차를 출발점에 두고, 그 각 경로에서 발화예정 트랜지션계열을 각각 구한다.

스텝 Sf 8에서는 각 무인차의 완료시각을 -1로 초기설정한다.

스텝 Sf 9에서는 발화 트랜지션계열을 비워서 초기 설정한다. 이 발화 트랜지션계열은 실제로 발화를 하는 트랜지션의 계열이고, 반드시 발화 예정 트랜지션계열과 일치하지 않는다.

스텝 Sf 10에서는 각 무인차에 대하여 완료시각이 현재시각과 같은가를 조사하고, 같은 경우에는 다음의 스텝 Sf 11로 진행하고, 같지않은 경우에는 스텝 Sf 12로 분기한다.

스텝 Sf 11에서는 스텝 Sf 10의 조건을 만족하는 모든 무인차의 발화예정 트랜지션에서 선두의 트랜지션을 취출하고, 발화 완료처리를 한다.

스텝 Sf 12(제 17도)에서는 각 무인차에 대하여 완료시각이 현재시각 이전인가를 조사하고, 이전인 경우에는 다음의 스텝 Sf 13으로 진행하고, 이전이 아닌 경우에는 스텝 Sf 20으로 분기한다.

스텝 Sf 13에서는 발화예정 트랜지션의 선두 트랜지션이 발화가능 한가를 조사하고, 발화가능한 경우에는 스텝 Sf 14로 진행하고, 발화 가능하지 않은 경우에는 스텝 Sf 17로 분기한다.

스텝 Sf 14에서는 스텝 Sf 13에서 발화가능으로 된 모든 트랜지션을 취출하여 발화처리를 한다.

스텝 Sf 15에서는 완료시각에 발화한 트랜지션의 이동시간을 가산하고, 완료시각을 갱신한다.

스텝 Sf 16에서는 스텝 Sf 14에서 발화처리를 했던 트랜지션을 각각 대응하는 발화 트랜지션계열로 등록(추가)하고, 스텝 Sf 20으로 진행한다.

한편, 스텝 Sf 17에서는 발화 불가능한 트랜지션에 대응하는 방해하는 무인차에 이동가능한 인접노드가 있는가를 조사한다. 즉 방해하는 무인차를 다른 노드로 추출시킬까를 조사한다. 이 결과, 추출가능한 경우에는 다음의 스텝 Sf 18로 진행하고, 불가능한 경우에는 스텝 Sf 20으로 분기한다.

스텝 Sf 18은 스텝 Sf 17에서 추출가능으로 된 무인차가 대기중에 있는가를 조사하고, 대기중인 경우에는 다음의 스텝 Sf 19로 진행하고, 대기중이 아닌 경우에는 스텝 Sf 20으로 분기한다.

스텝 Sf 19에서는 스텝 Sf 17 및 18의 조건을 만족하는 무인차에 대한 퇴피경로처리(제 6도, 스텝 Sa 1)를 하고, 퇴피경로를 구한다.

스텝 Sf 20에서는 모든 무인차에 대응하는 발화 트랜지션의 완료시각이 현재시각 이전에 있는가를 조사하고, 이전이면 스텝 Sf 22로 분기하고, 이전이 아니라면 스텝 Sf 21로 진행한다.

스텝 Sf 21에서는 모든 무인차 중에서 가장 가까운 미래의 완료시각을 가지는 무인차를 결정하고, 그 완료시각을 현재시각으로 결정한다. 그래서, 스텝 Sf 10(제 16도)로 돌아간다.

스텝 Sf 22에서는 모든 무인차의 발화예정 트랜지션계열이 비어있는가를 조사하고, 비어 있으면 스텝 Sf 32(제 18도)로 분기하고, 비어있지 않으면 다음의 스텝 Sf 23으로 진행한다.

스텝 Sf 23에서는 전술한 데드락 파악처리(제 8도, 스텝 Sb 1)에 의해 데드락의 상황을 조사한다.

스텝 Sf 24에서는 스텝 Sf 23의 처리에서 얻어진 경합루우프에 기하여 상술한 발화순서 조정처리(제 10도, 스텝 Sc 1)에 의해 각 트랜지션의 발화순서를 조정한다.

스텝 Sf 25에서는 앞 스텝 Sf의 발화순서의 조정이 성공했는가를 조사하고, 실패한 경우에는 다음의 스텝 Sf 26으로 진행하고, 성공한 경우 즉 데드락이 해소된 경우에는 스텝 Sf 6(제 16도)로 돌아간다.

스텝 Sf 26에서는 우회경로탐색처리(제 12도, 스텝 Sd1)에 의해 우회경로를 탐색한다.

스텝 Sf 27에서는 앞 스텝에서의 우회경로탐색이 성공했는가를 조사하고, 실패한 경우에는 다음의 스텝 Sf 28로 진행하고, 성공한 경우에는 스텝 Sf 6(제 16도)를 돌아간다.

스텝 Sf 28에서는 대피경로탐색(제 14도, 스텝 Se 1)에 의해 대피경로를 탐색한다.

스텝 Sf 29에서는 앞 스텝의 대피경로탐색이 성공했는가를 조사하고, 실패한 경우에는 다음의 스텝 Sf 30으로 진행하고, 성공한 경우에는 스텝 Sf 6(제 16도)로 돌아간다.

스텝 Sf 30에서는 현재의 시행횟수를 조사하고, 그것이 「1」인 때에는 다음의 스텝 Sf 31로 진행하고, 「1」이 아닌 경우에는 동작계획실패로 모든 처리를 종료한다(스텝 Sf 34).

스텝 Sf 31에서는 시행횟수를 2로 증가한 후 스텝 Sf 3(제 16도)로 돌아간다.

스텝 Sf 32에서는 무인차의 동작계획이 성공한 경우에 실행되고, 현재의 경로를 무인차의 최종경로로 설정한다.

스텝 Sf 33에서는 발화 트랜지션계열을 기본으로 각 노드를 정류하는 무인차의 순서(노드 예약시퀀스)를 작성하고, 모든 처리를 종료한다(스텝 Sf 34).

VIII. 패쇄트리네트 시뮬레이션의 처리에

다음에, 이상의 패쇄트리네트 시뮬레이션의 구체예를 상술한 제 4도의 실행도(a)·경로(b)의 조건에 있어서 설명한다. 제 19도는 시뮬레이션의 과정을 도시한 패쇄트리네트도이다.

먼저, 초기 상태 {제 19도(a)} 에서 우선 발화처리(스텝 Sf 14)에 의한 무인차 #1에 대응하는 트랜지션 T 23이 무인차 #2에 대응하는 트랜지션 T 67이 각각 발화한다. 이들의 완료시각은 각각 1·2초로 된다. 그래서, 플레이스 P3 및 P4에 흰마킹이 놓여진다.

다음에, 스텝 Sf 21에서의 현재 시각이 무인차 #1의 완료시각 즉 1초로 갱신되고, 스텝 Sf 10으로 돌아간다. 스텝 Sf 11의 발화완료처리에 의해 플레이스 P 2로부터 검은 마킹이, 플레이스 P 3으로부터 하얀 마킹이 제거되고, 플레이스 P 3에 검은 점이 표시된다 {제 19도(b)}. 이 상태는 무인차 #1이 노드 3으로 도착한 것을 의미한다. 여기서, 발화중의 트랜지션 T67은 장방향으로 둘러싼다.

다음에, 스텝 Sf 14에 의한 무인차 #1에 대응하는 트랜지션 T34가 발화하고, 스텝 Sf 15에서 무인차 #1의 완료시각이 4초로 세트된다. 다음, 스텝 Sf 21에서 현재시각이 무인차 #2의 완료시각의 3초로 갱신되고, 스텝 Sf 10을 경유하여 다시 돌아온 스텝 Sf 11에서 트랜지션 T67의 발화완료처리가 행하여진다.

스텝 Sf 13에서 무인차 #2의 트랜지션 T73이 조사되지만 출력선의 플레이스 P3에 검은 마킹이 있기 때문에 발화 불가능하다. 이것은 무인차 #1이 노드 3을 점류하고 있기 때문이다. 무인차 #1은 현재노드 3에서 4로 이동중이기 때문에 추출시킬 수 없다. 이 때문에, 무인차 #2와 노드 7에서 대기하게 된다. 스텝 Sf 21에서 무인차 #1의 완료시각 즉 4초로 현재시각이 갱신된다.

스텝 Sf 14에서 트랜지션 T34의 발화처리가 행하여져 무인차 #1이 목표노드 4에 도착한다. 다음, 무인차 #1이 노드 3을 개방했기 때문에 스텝 Sf 14에서 트랜지션 T73의 발화처리가 행해진다 {제 19도(d)}.

트랜지션 T73의 발화처리가 끝난 시점 {제 20도(a)} 에서 스텝 Sf 14에 의해 최후의 트랜지션 T34를 발화시키도록 하지만, 발화 불가능하기 때문에 무인차 #1에 추출을 시킨다.

되피경로탐색처리(스텝 Sf 19)에 의해 무인차 #1의 되피경로(노드 4→8)가 구해지는 대응하는 트랜지션 T48이 발화하여야 할 트랜지션에 가해져 발화한다.

이하, 동일한 처리로 시뮬레이션이 진행되고, 제 20도(c)까지 진행하면 발화하여야 할 트랜지션이 없게 되고, 스텝 Sf 32의 처리에 들어간다.

트랜지션의 발화계열 = {T23(#1)·T67(#2)·T34(#1)·T73(#2)·T48(#1)·T34(#2)}로부터 각 노드를 점유하는 무인차의 선행관계를 조사하고, 제 21도(a)에 도시한 바와같은 노드 예약 시퀀스를 작성한다. 또한, 이 노드 예약시퀀스로부터 본도(b)에 도시한 바와 같은 운행계획도가 작성된다. 본도에 있어서, 무인차 #1 및 #2는 실선과 파선에 각각 대응하고 있다.

또한, 화살표는 각 무인차의 이동을 나타내고, 수평선은 노드가 예약되어 있는 기간을 나타낸다. 예를들면, 무인차 #1은 노드 2에서 1초로 노드 3으로 이동하고, 더욱 3초로 노드 4로 이동을 한다. 이사이 노드 3은 시각 0에서 4까지 무인차 #1에 예약된다.

D : 전체의 동작에 1

이하에서, 제 28도에 도시한 반송로(101)에서의 운행관리제어장치(102)(제 1도)의 동작을 설명한다. 이하의 도면에서 제 28도와 대응하는 부분에는 동일한 부호를 붙이고, 그 설명을 생략한다. 또한, 이 동작예에서 출발점 및 목표점을 제 22도(a)에 도시한다.

먼저, 경로계획부(109)는 경로탐색부(110)에 대해 탐색지시 {제 22도(a)}를 출력한다. 경로탐색부(110)는 이 탐색지시에 따라서 각 무인차 #1 내지 #5의 반송경로(초기경로)를 탐색하고, 그 결과인 초기경로를 경로계획부(109)로 출력한다. 제 23도(a)는 초기경로를 도시한 실행도이고, 본도에서의 무인차 #1 내지 #7의 경로는 각각 점선·긴 일점쇄선·이점쇄선·일점쇄선·파선·실선·긴 파선으로 표시되어 있다.

경로계획부(109)는 이 초기경로에서의 노드 2·3사이, 노드 4~6 사이 및 노드 8~10 사이가 역방향 구간으로 되어있기 때문에 코스트에 대응하여 주행로의 특정구간의 방향결정을 하고, 다시 경로탐색부(110)에 탐색지시를 출력한다. 이상의 동작이 역방향 구간이 없게 될때까지 행해지고, 제 22도(b) 및 제 23도(b)에 도시한 바와같이 기본경로가 얻어진다. 경로계획부(109)는 이 기본경로를 동작계

획부(108)로 출력한다.

동작계획부(108)는 이 기본경로에 기하여 상술한 동작계획처리(제 16도 · 17도 · 18도)를 한다. 또한, 이 처리의 사이에 행해지는 우회경로등의 경로탐색은 경로계획부(109)를 개재하여 경로탐색부(110)에서 행해진다. 또한, 이 경우 출발노드 · 목표노드 및 통행금지구간은 동작계획부(108)로부터 출력된다.

이상의 처리에 의해 제 22도(c) 및 제 23도(c)에 도시한 바와같이 최종경로가 얻어진다. 이 최종경로에서는 기본경로 {제 22도(b)}에 대하여 무인차 #1의 퇴피경로(노드 20→ 6)가 추가되어 있다. 또한, 제 24도는 이때의 무인차 #1 내지 #7의 각각의 이동을 시간적으로 도시한 운행계획도이다.

E : 전체의 동작에 2

다음에, 상술한 반송로(101)의 노드 20 · 21사이의 통행금지로 되는 경우의 동작에 대하여 설명한다. 단, 이 동작에에서 각 무인차 #1 내지 #7의 현재치 및 목표치는 상술한 동작에 1과 동일하다 {제 22도(a)}. 또한, 이 경우 노드 6 · 7사이, 노드 7 · 8 사이, 노드 21 · 22 사이는 이를 묶은 경로 이외에 우회하는 경로가 존재하지 않기 때문에 역방향구간에 포함되지 않는다.

여기서도 동작에 1과 동일한 처리가 행해지고, 먼저, 경로탐색부(110)에서의 제 26도(a)의 운행도에 도시한 바와같은 초기경로가 얻어진다. 다음에 경로계획부(109)에 의해 제 25도(b) 및 제 26도(b)에 도시한 바와같은 기본경로가 작성된다. 이들 도면에서의 초기경로 {제 26도(a)}에 있었던 역주행구간은 없어지게 된다.

그래서, 동작계획부(108)에서의 제 25(c) 및 제 26도(c)에 도시한 바와같은 최종경로가 작성된다. 이 최종경로에서는 기본경로 {제 26도(b)}에 대하여 무인차 #1의 퇴피경로(노드 20→ 6) 및 무인차 #5의 퇴피경로(노드 8→ 22→ 23→ 24→ 10→ 9)가 추가되어 있다. 또한, 제 27도는 이때의 운행계획도이고, 본도에서의 무인차 #1 내지 #7의 노드예약은 제 22도와 동일한 종류의 선으로 도시되어 있다.

이상 설명한 바와같이 본 발명에 의하면, 무인차가 이동을 개시하기 전에 미리 무인차끼리의 간섭을 고려하여 모든 무인차의 주행경로 및 주행순서를 얻을 수 있기 때문에 다수의 무인차가 주행경로상에서 빈번히 간섭하는 가능성이 있는 경우에는 원활한 이동이 가능하게 되고, 따라서 무인차의 반송효율을 향상시킬 수 있다는 효과가 얻어진다.

(57) 청구의 범위

청구항 1

정지위치에 있는 복수의 노드와, 상기 노드간을 접속하는 접속로로 되는 주행로를 주행하는 복수 무인차의 운행을 제어하는 운행관리제어장치에 있어서,

동작계획부(108) · 경로계획부(109) 및 경로탐색부(110)로 이루어지는 제어부(107)에 의해,

상기 경로계획부(109)가 최초 경로탐색을 지시하고, 탐색된 경로중 역방향 구간이 없을 때까지 탐색지시를 하고, 상기 경로탐색부(110)가 상기 경로계획부(109)의 지시에 따라 무인차의 반송로를 탐색하여 상기 경로계획부(109)로 결과를 출력하는 제1수단과,

상기 제1수단에 의해 탐색된 각 무인차의 주행경로에 기하여 상기 동작계획부(108)로 각 무인차의 시간적 이동을 시뮬레이션하고, 어느 것인가 무인차의 진행불능을 검출한 경우에, 노드 통과순서변경 · 우회 경로탐색 · 대피경로탐색의 어느 것인가의 방법으로 상기 무인차의 진행불능을 해제하는 제2수단과,

를 구비하여 되는 것을 특징으로 하는 운행관리제어장치.

청구항 2

정지위치에 있는 복수의 노드와, 상기 노드간을 접속하는 접속로로 되는 주행로를 주행하는 복수 무인차의 운행을 제어하는 운행 관리제어방법에 있어서,

각 무인차가 동일 접속로를 서로 역방향 주행하지 않도록 각 무인차의 주행경로를 탐색하는 제 1스텝과,

상기 제 1스텝에 의해 얻어진 주행경로에 기하여 각 무인차의 시간적인 이동을 조사하고, 어느것인가 무인차의 진행불능이 검출된 경우에는 이미 주행을 종료하고 있는 무인차의 경로에 퇴피경로를 추가하는 제 2스텝과,

상기 제 2스텝에서의 진행불능이 해소될수 없는 경우에 무인차의 주행 순서를 변경하는 제 3스텝과,

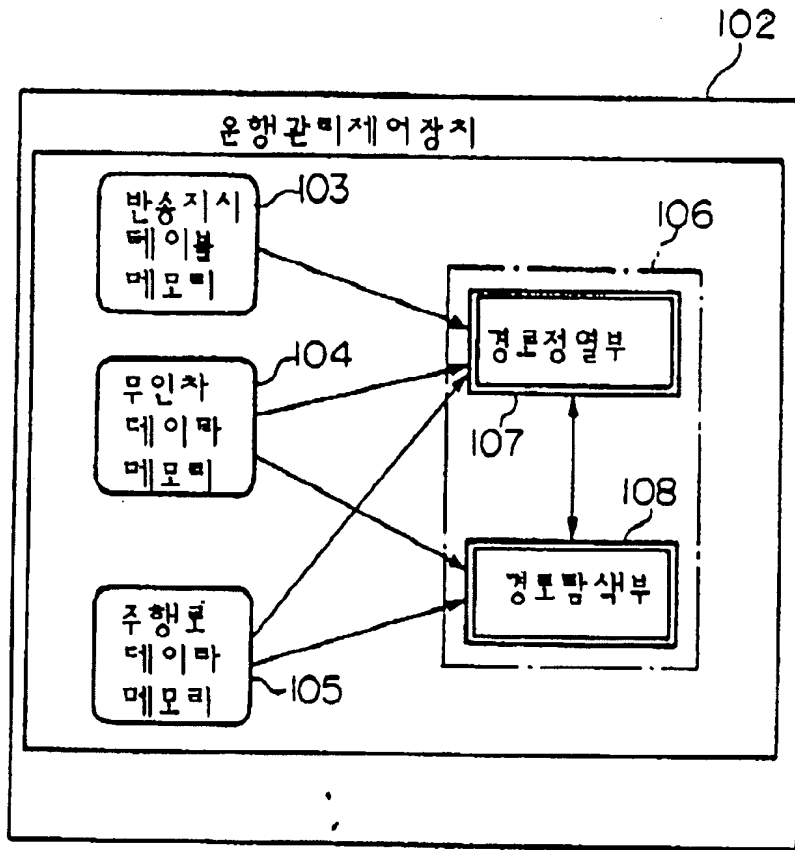
상기 제 3스텝에서의 진행불능이 해소될수 없는 경우에 무인차의 경로에 우회경로를 추가하는 제 4스텝과,

상기 제 4스텝에서의 진행불능이 해소될수 없는 경우에 무인차의 경로에 대피경로를 추가하는 제 5스텝과,

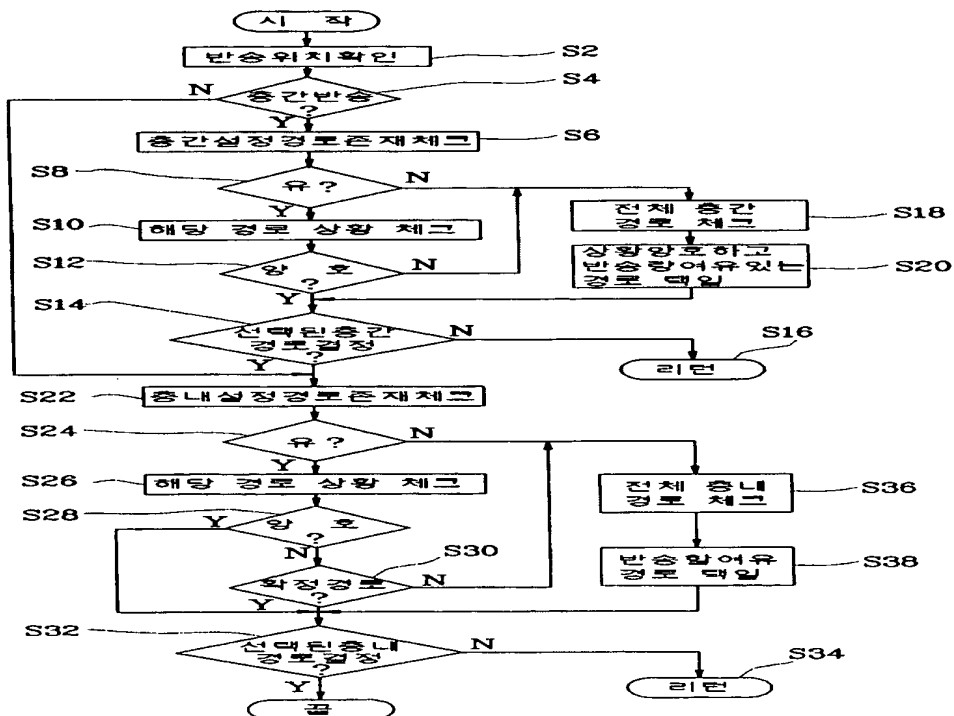
를 가지는 것을 특징으로 하는 운행관리제어방법.

도면

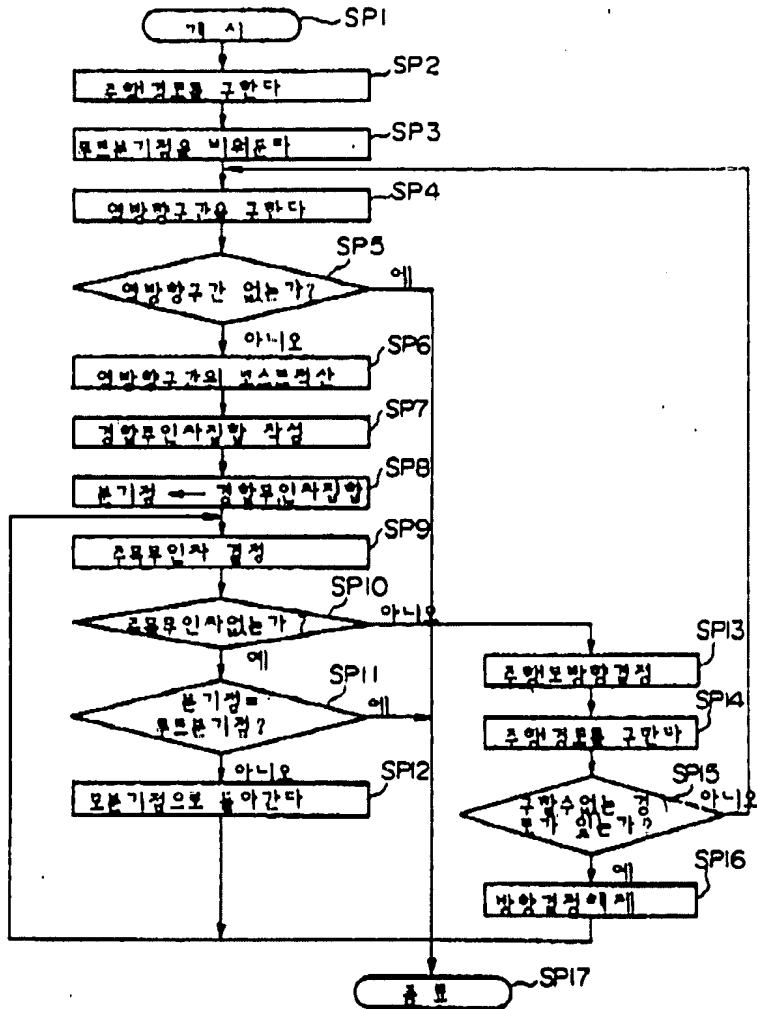
도면1



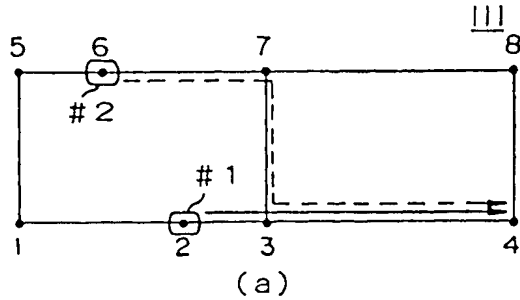
도면2



도면3



도면4



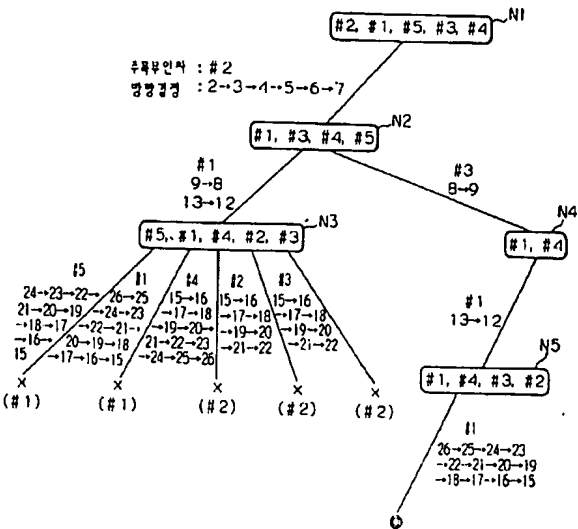
무인차	출발점	목표점	기본경로 (이동시간: 초)
# 1	2	4	2 [1] → 3 [3] → 4
# 2	6	4	6 [2] → 7 [4] → 3 [3] → 4

(b)

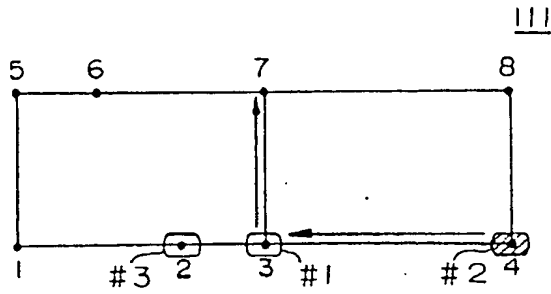
도면5

무인차	출발점	목표점	기본경로 (이동시간: 초)
# 1	2	4	2 [1] → 3 [3] → 4
# 2	6	4	6 [2] → 7 [4] → 3 [3] → 4

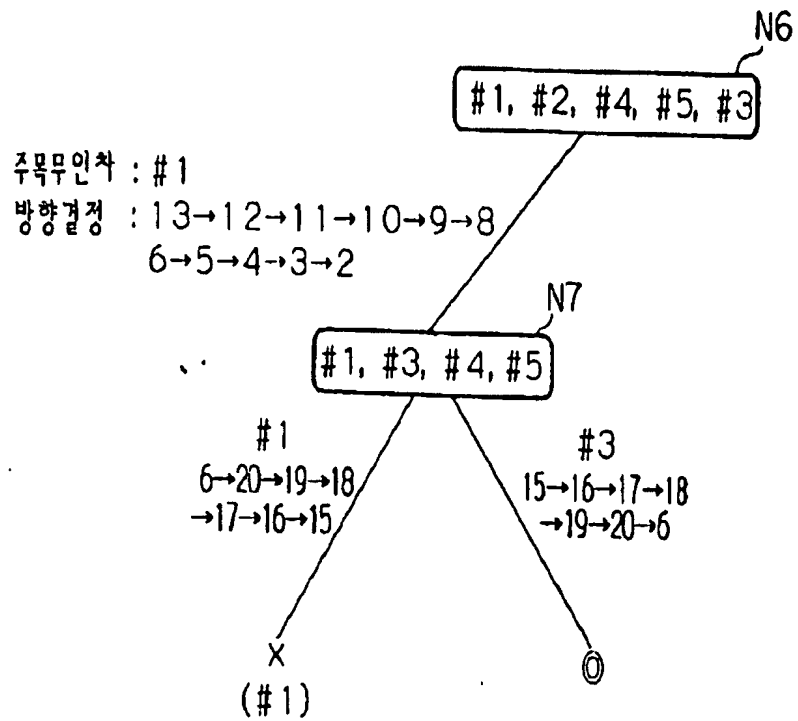
도면6



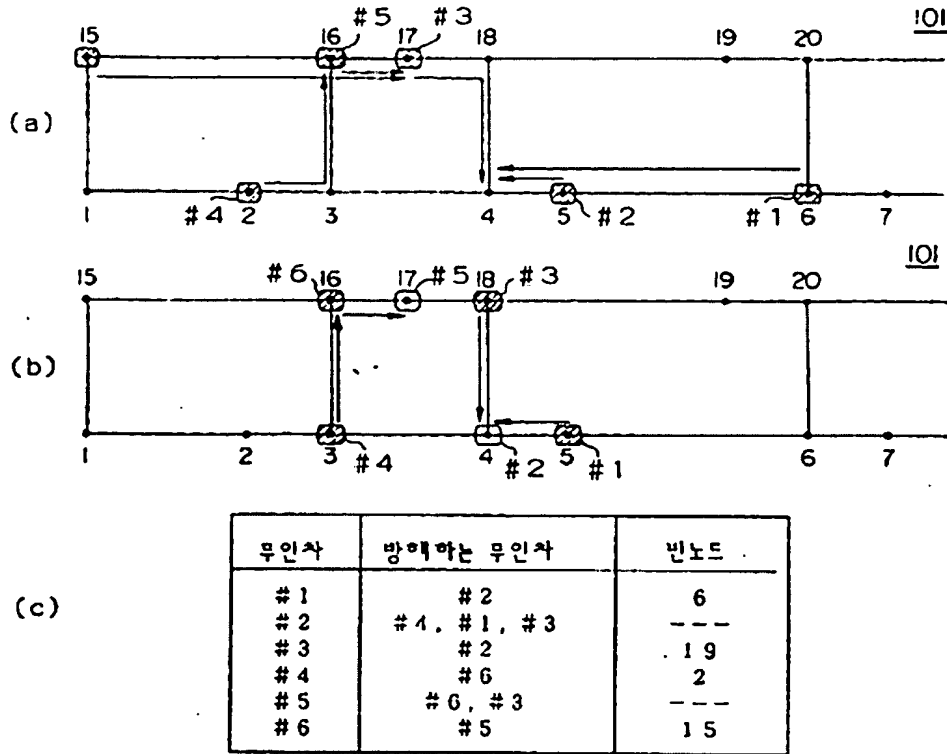
도면7



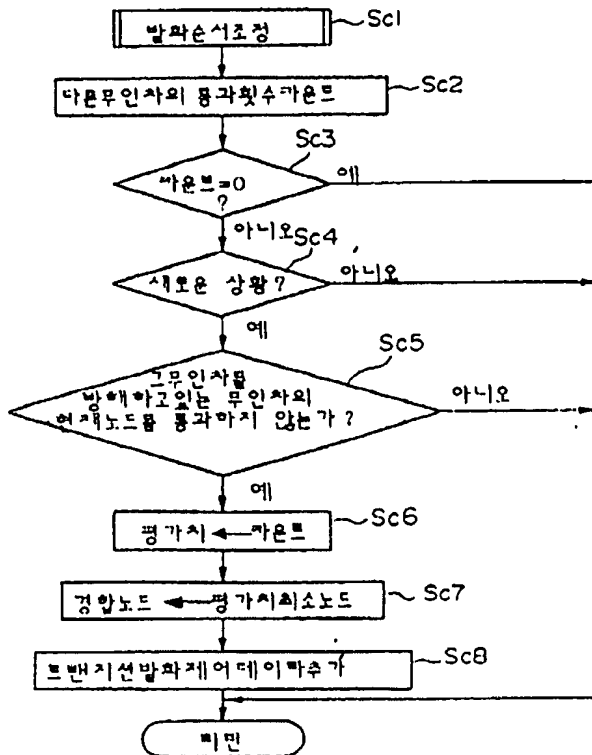
도면8



도면9



도면10



도면 11

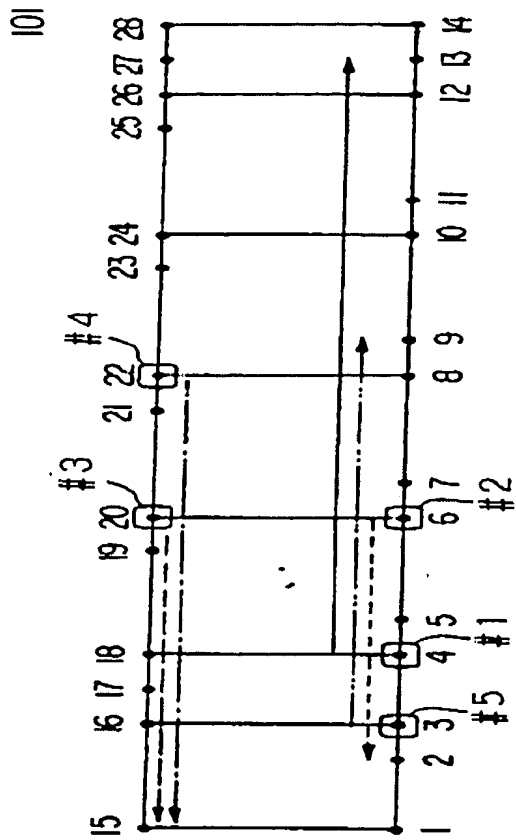
노드	경로
#1	6→5→(4)
#2	5→(4)→3→16
#3	17→18→(4)→3
#4	2→3→(16)→17→18
#5	16→(17)→18→19
#6	15→16→17

(a)

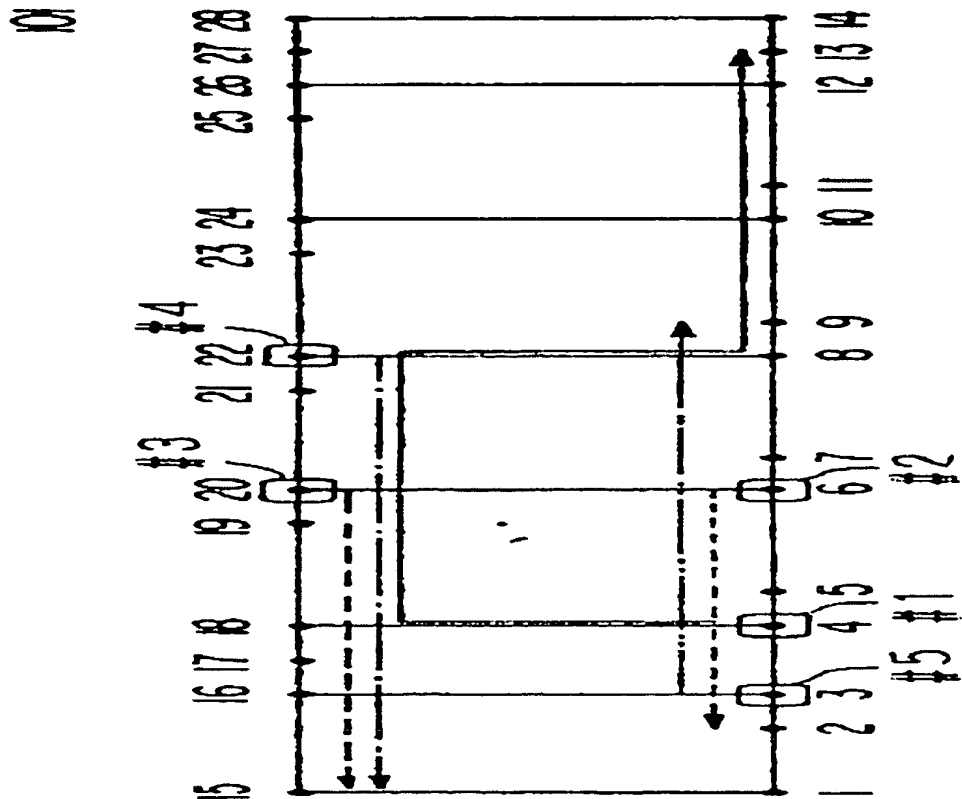
노드	노드예외 시퀀스
2	#4
3	#4, #2, #3
4	#2, #3, #1
5	#2, #1
6	#1
15	#6
16	#5, #4, #6, #2
17	#3, #5, #4, #6
18	#3, #5, #4
19	#5

(b)

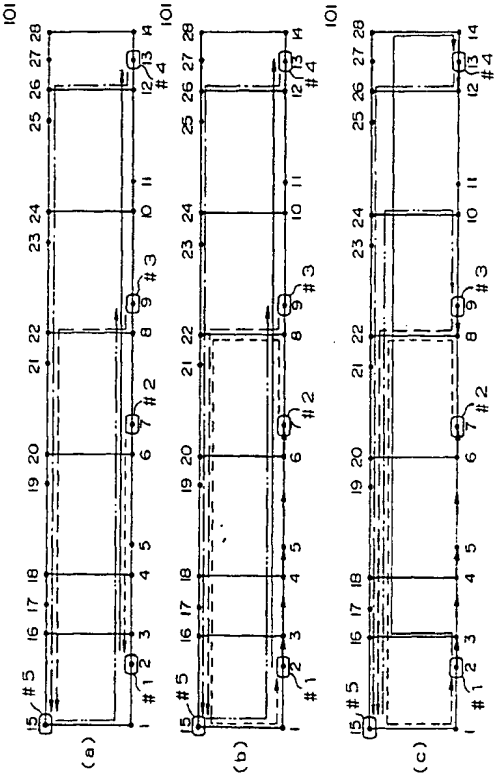
도면 12



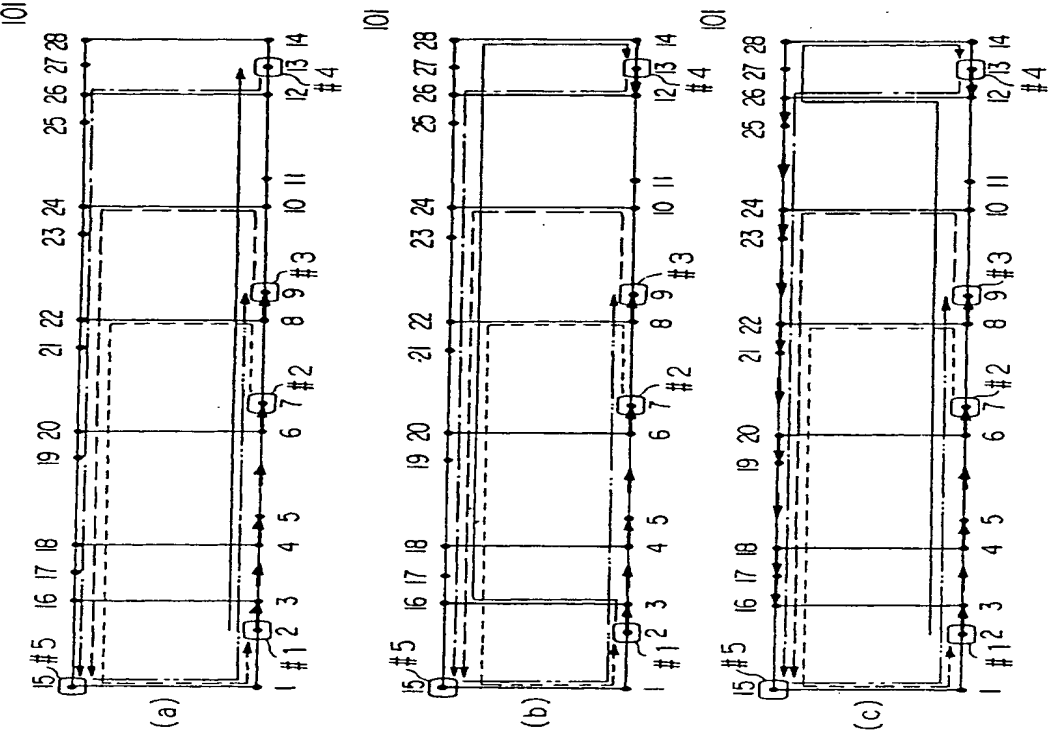
도면 13



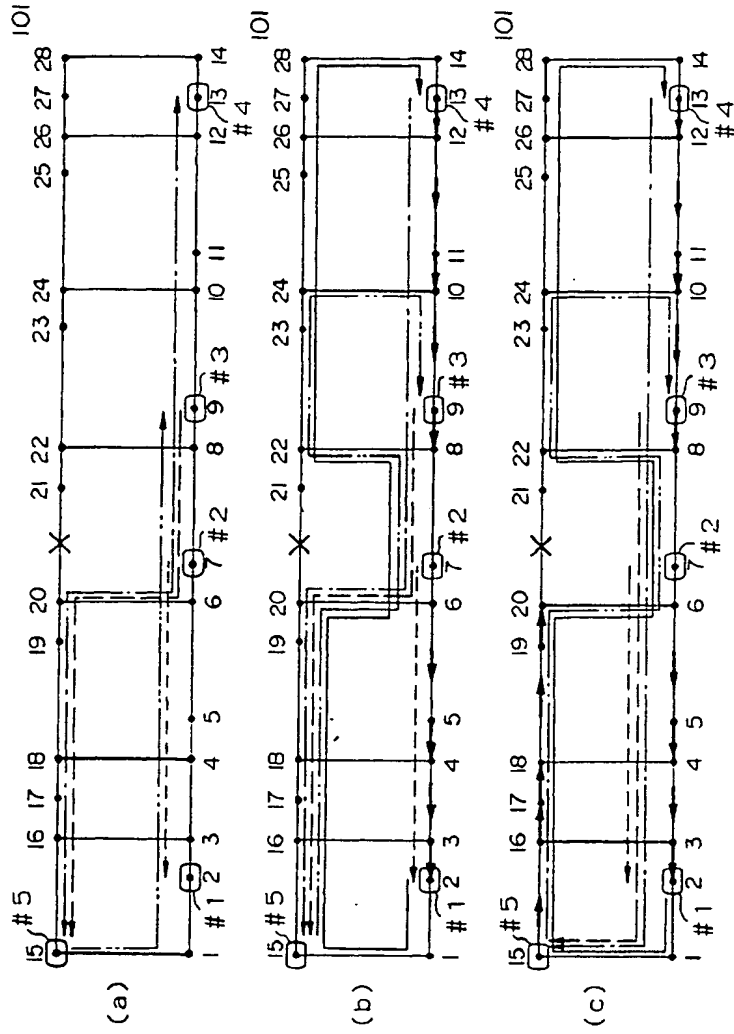
도면 14



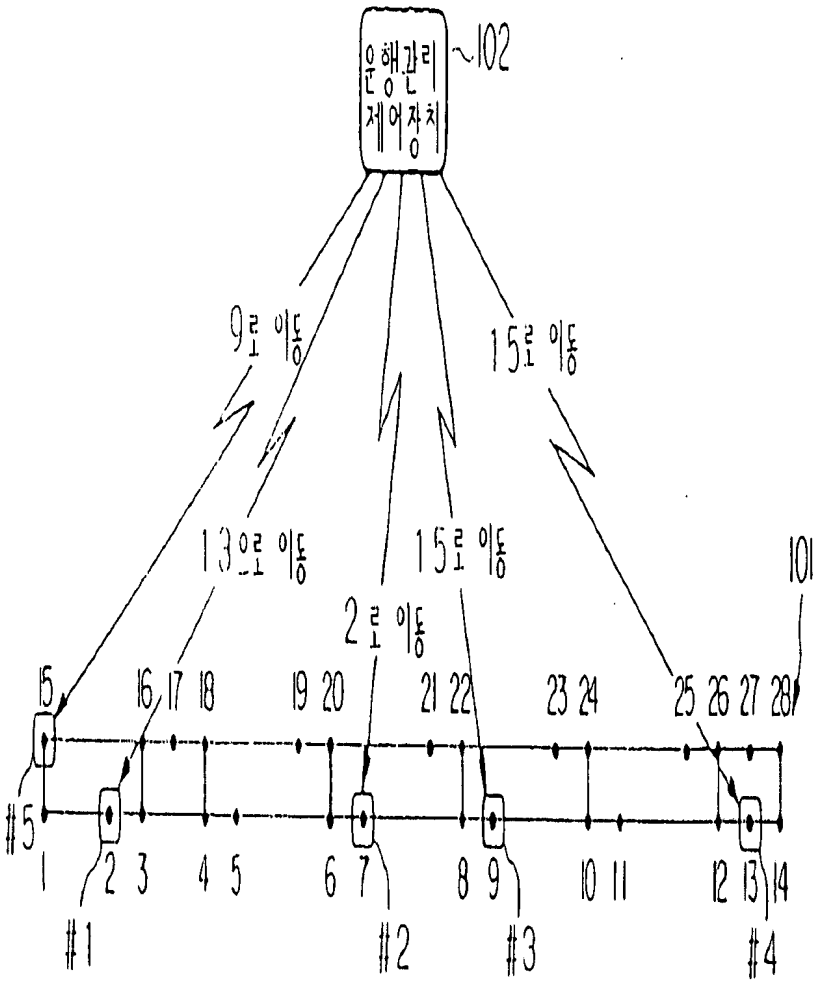
도면 15



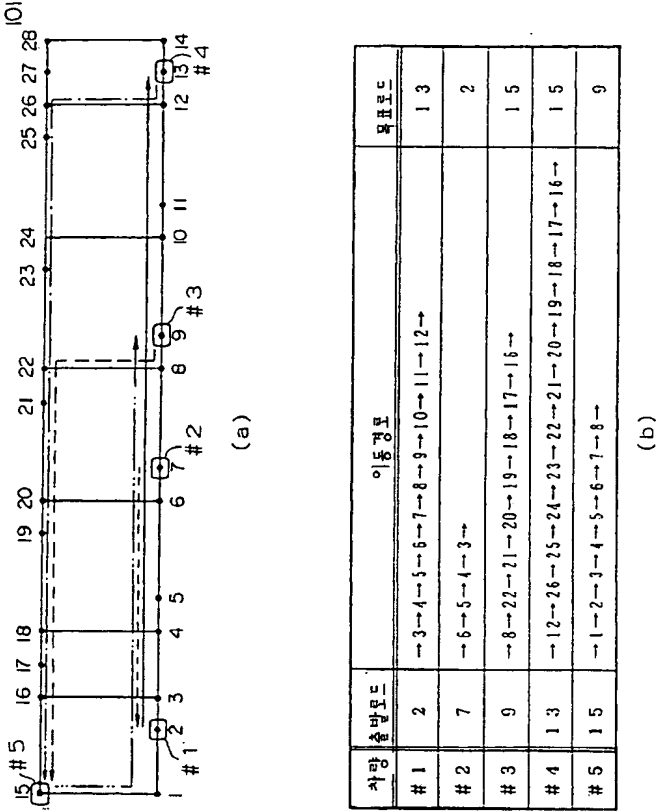
도면 16



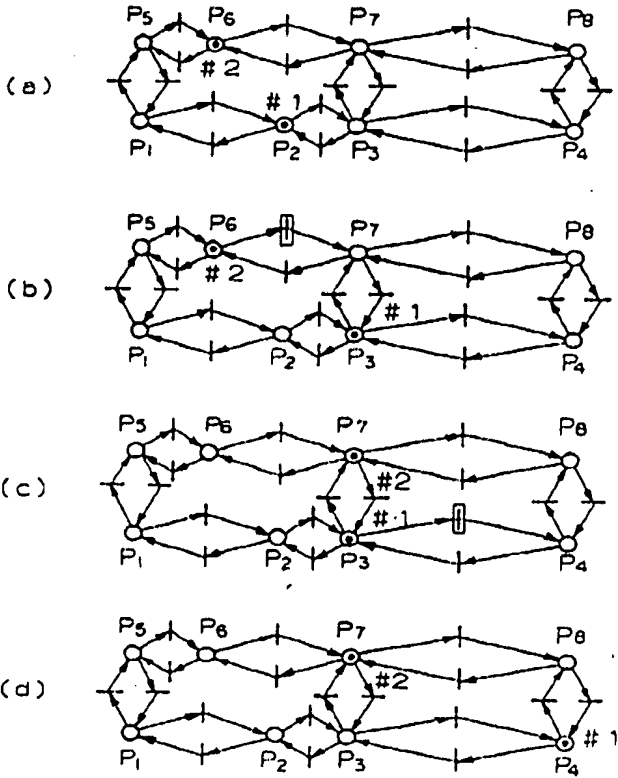
도면 17



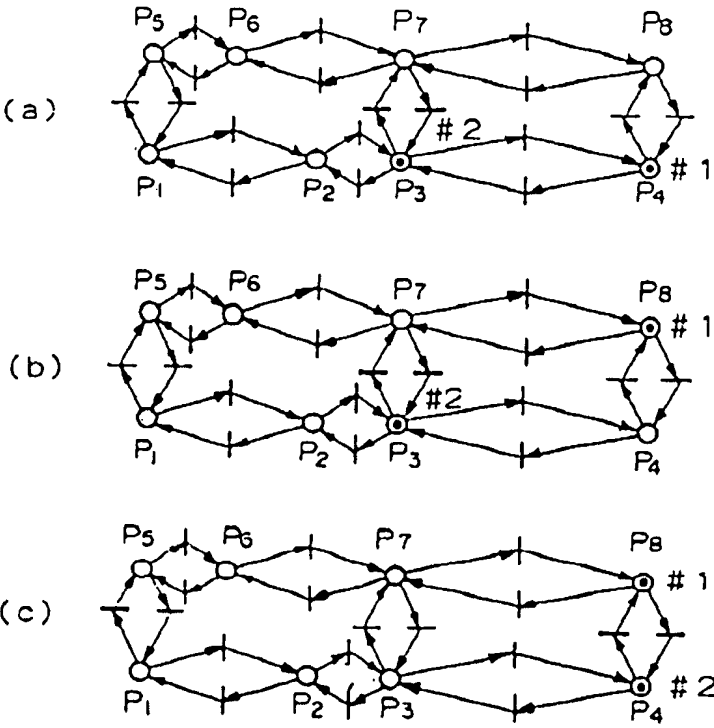
도면 18



도면 19



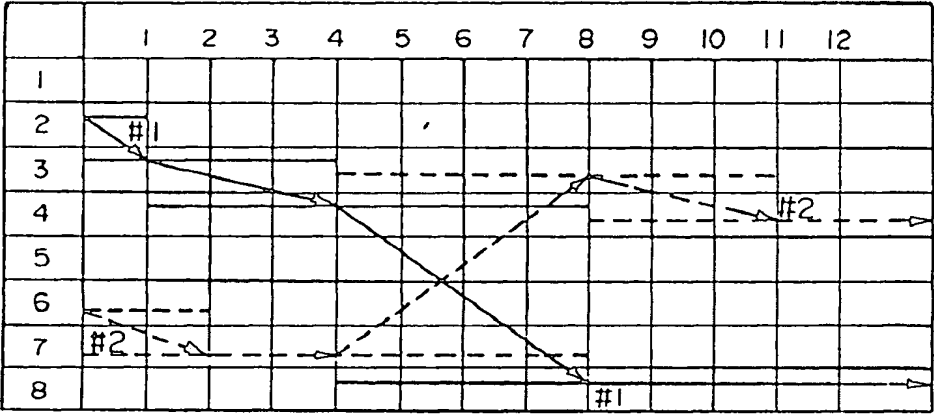
도면20



도면21

노드	예약무인차
2	# 1
3	# 1 , # 2
4	# 1 , # 2
6	# 2
7	# 2
8	# 1

(a)



(b)

도면22

(a)

무인차	출발점	목표점
#1	19	20
#2	21	26
#3	3	20
#4	17	10
#5	11	8
#6	2	19
#7	20	1

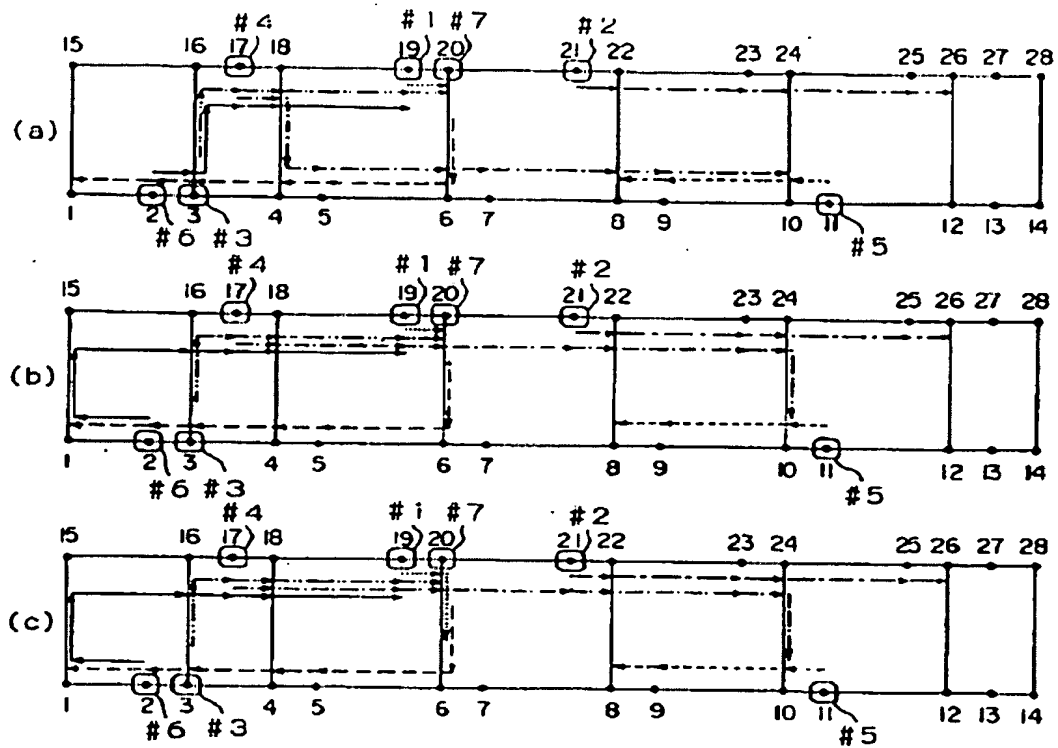
(b)

무인차	기본경로
#1	19→20
#2	21→22→23→24→25→26
#3	3→16→17→18→19→20
#4	17→18→4→5→6→7→8→9→10
#5	11→10→9→8
#6	2→1→15→16→17→18→19
#7	20→6→5→4→3→2→1

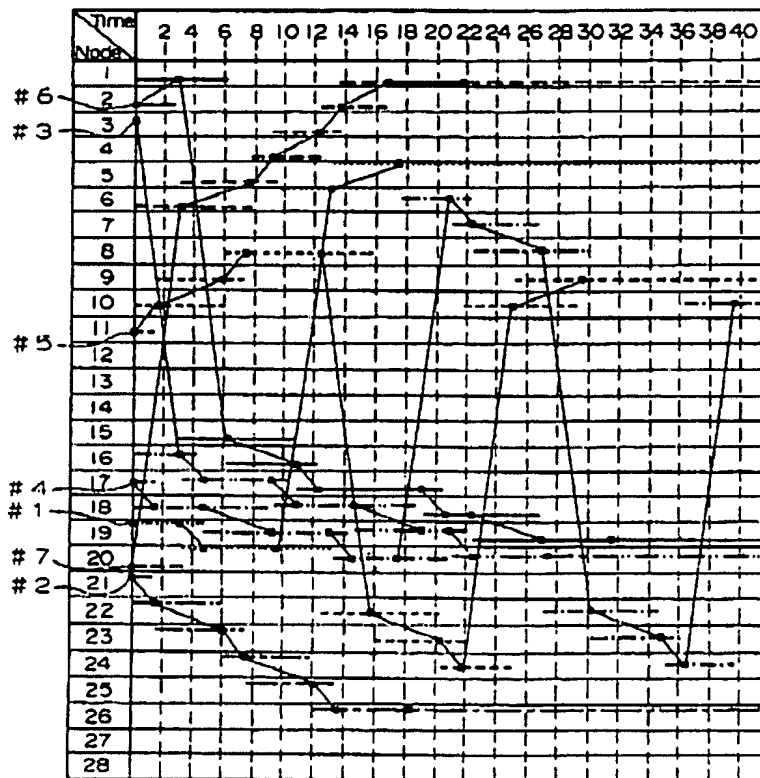
(c)

무인차	최종경로
#1	19→20→6
#2	21→22→23→24→25→26
#3	3→16→17→18→19→20
#4	17→18→4→5→6→7→8→9→10
#5	11→10→9→8
#6	2→1→15→16→17→18→19
#7	20→6→5→4→3→2→1

도면23



도면24



도면25

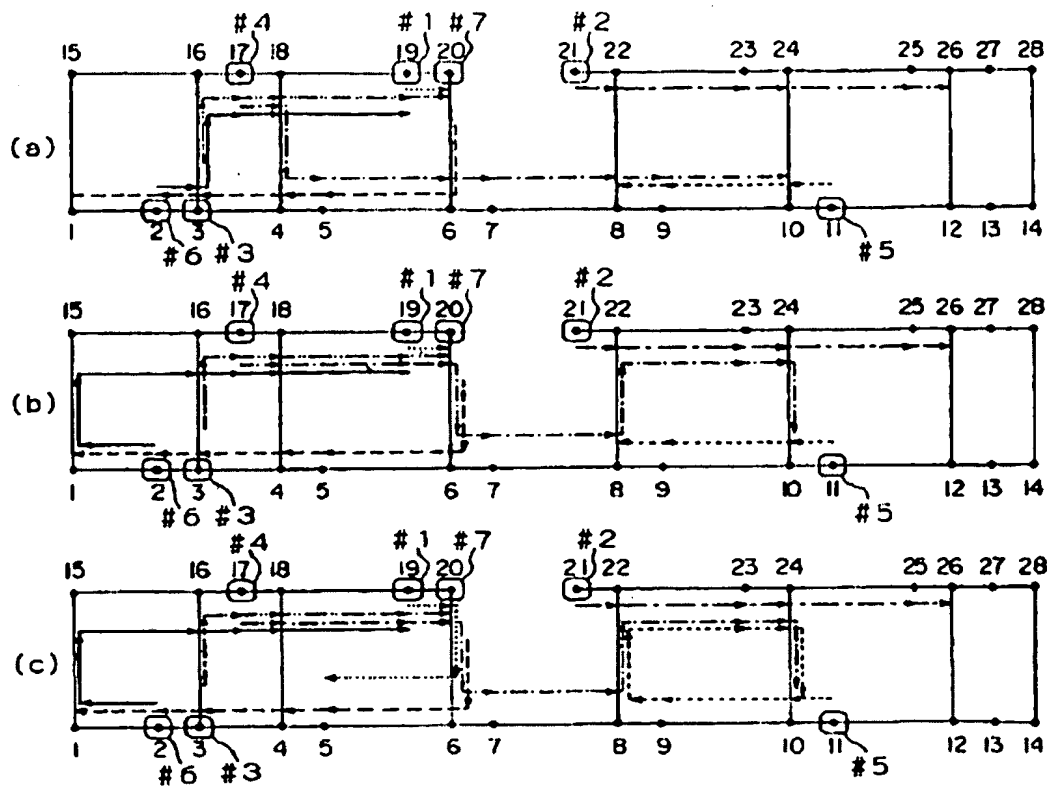
무인차	기본 경로
# 1	19→20
# 2	21→22→23→24→25→26
# 3	3→16→17→18→19→20
# 4	17→18→19→20→6→7→8→22→23→24→10
# 5	11→10→9→8
# 6	2→1→15→16→17→18→19
# 7	20→6→5→4→3→2→1

(a)

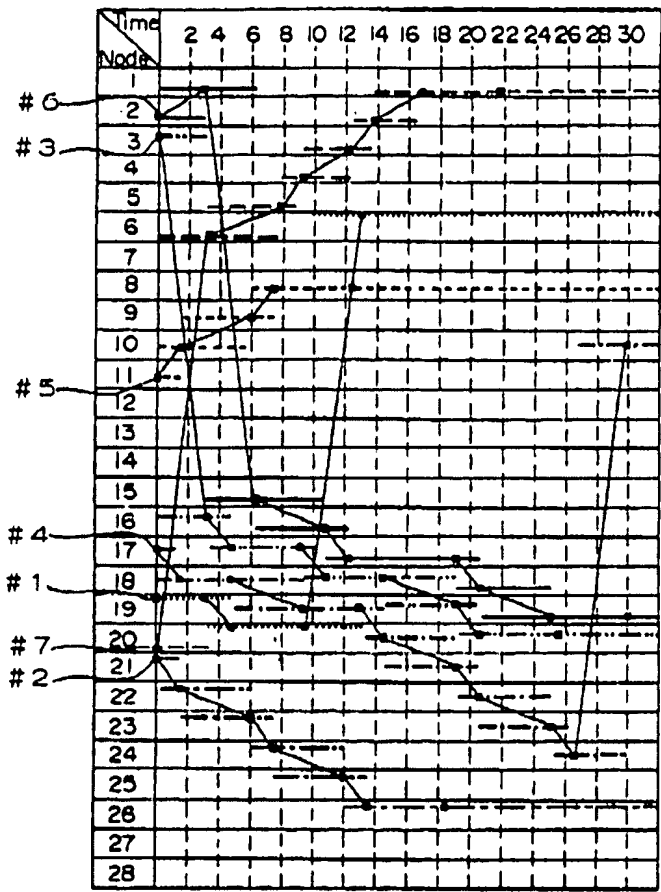
무인차	최종 경로
# 1	19→20→6→5
# 2	21→22→23→24→25→26
# 3	3→16→17→18→19→20
# 4	17→18→19→20→6→7→8→22→23→24→10
# 5	11→10→9→8→ <u>22→23→24→10→9</u>
# 6	2→1→15→16→17→18→19
# 7	20→6→5→4→3→2→1

(b)

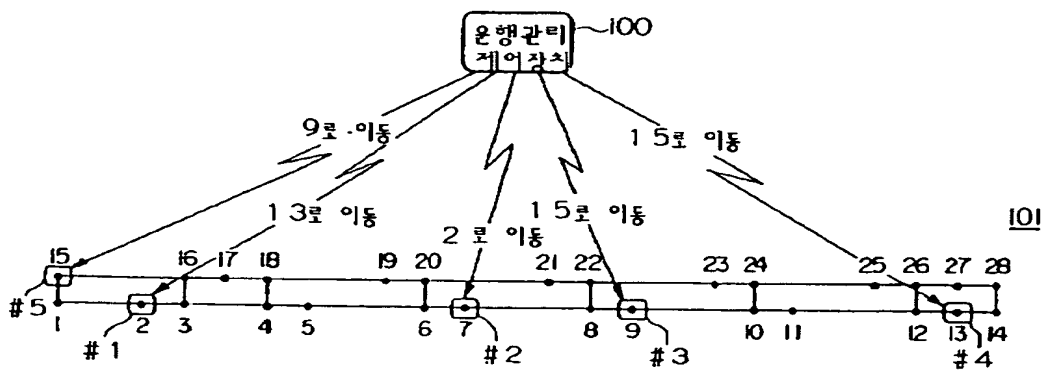
도면26



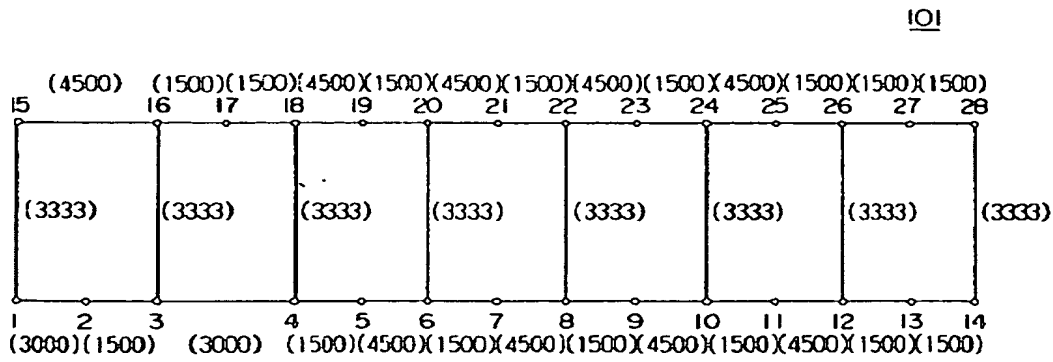
도면27



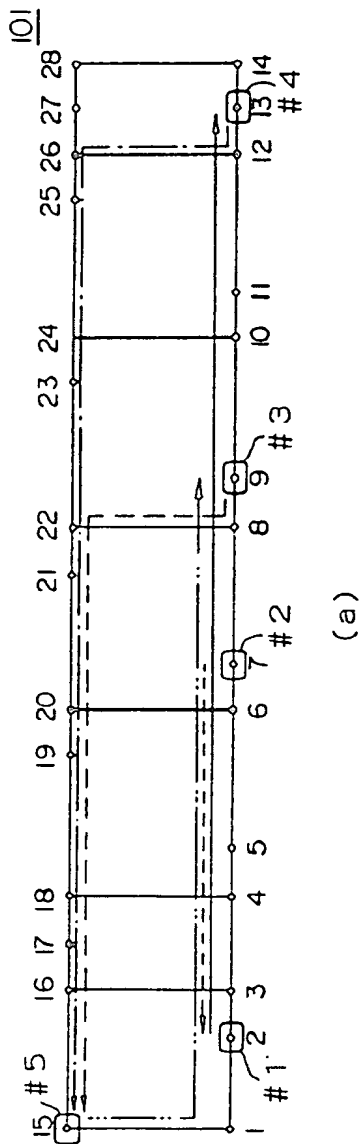
도면28



도면29



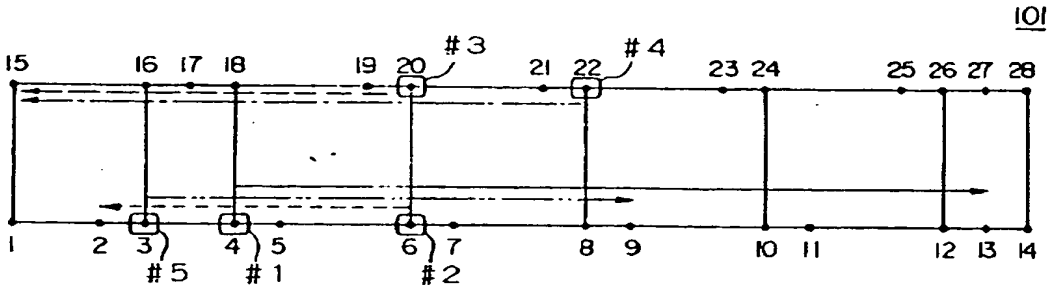
도면30



차량	출발노드	이동경로	목적노드
# 1	2	→ 3 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11 → 12 →	1 3
# 2	7	→ 6 → 5 → 4 → 3 →	2
# 3	9	→ 8 → 22 → 21 → 20 → 19 → 18 → 17 → 16 →	1 5
# 4	1 3	→ 12 → 26 → 25 → 24 → 23 → 22 → 21 → 20 → 19 → 18 → 17 → 16 →	1 5
# 5	1 5	→ 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 →	9

(b)

도면31



도면32

